Technical Note

TinyM0 T12M 应用开发指南

V1.00 Date:2010/12/23 工程技术笔记

类别	内容
关键词	LPC1200、Cortex-M0、TinyM0 T12M、TKScope CK100 仿真器、TKStudio、K-Flash、TKS_COM
摘　要	TinyM0 T12M 应用开发指南

广州致远电子有限公司

ZLG®

修订历史

版本	日期	原因
V1.00	2010/12/23	发布文档

销售与服务网络(一)

广州周立功单片机发展有限公司



地址：广州市天河北路 689 号光大银行大厦 12 楼 F4
邮编：510630
电话：(020)38730916　38730917　38730972　38730976　38730977
传真：(020)38730925
网址：www.zlgmcu.com

广州专卖店

地址：广州市天河区新赛格电子城 203-204 室
电话：(020)87578634　87569917
传真：(020)87578842

南京周立功

地址：南京市珠江路 280 号珠江大厦 1501 室
电话：(025) 68123901　68123902
传真：(025) 68123900

北京周立功

地址：北京市海淀区知春路 113 号银网中心 A 座 1207-1208 室(中发电子市场斜对面)
电话：(010)62536178　62536179　82628073
传真：(010)82614433

重庆周立功

地址：重庆市石桥铺科园一路二号大西洋国际大厦(赛格电子市场)1611 室
电话：(023)68796438　68796439
传真：(023)68796439

杭州周立功

地址：杭州市天目山路 217 号江南电子大厦 502 室
电话：(0571)89719480　89719481　89719482
89719483　89719484　89719485
传真：(0571)89719494

成都周立功

地址：成都市一环路南二段 1 号数码科技大厦 403 室
电话：(028)85439836　85437446
传真：(028)85437896

深圳周立功

地址：深圳市深南中路 2070 号电子科技大厦 C 座 4 楼 D 室
电话：(0755)83781788(5 线)
传真：(0755)83793285

武汉周立功

地址：武汉市洪山区广埠屯珞瑜路 158 号 12128 室(华中电脑数码市场)
电话：(027)87168497　87168297　87168397
传真：(027)87163755

上海周立功

地址：上海市北京东路 668 号科技京城东座 7E 室
电话：(021)53083452　53083453　53083496
传真：(021)53083491

西安办事处

地址：西安市长安北路 54 号太平洋大厦 1201 室
电话：(029)87881296　83063000　87881295
传真：(029)87880865

销售与服务网络(二)

广州致远电子有限公司

地址：广州市天河区车陂路黄洲工业区 3 栋 2 楼

邮编：510660

传真：(020)38601859

网址：www.embedtools.com (嵌入式系统事业部)

www.embedcontrol.com (工控网络事业部)

www.ecardsys.com (楼宇自动化事业部)



技术支持：

CAN-bus：

电话：(020)22644381 22644382 22644253

邮箱：can.support@embedcontrol.com

iCAN 及数据采集：

电话：(020)28872344 22644373

邮箱：ican@embedcontrol.com

MiniARM：

电话：(020)28872684 28267813

邮箱：miniarm.support@embedtools.com

以太网：

电话：(020)22644380 22644385

邮箱：ethernet.support@embedcontrol.com

无线通讯：

电话：(020) 22644386

邮箱：wireless@embedcontrol.com

串行通讯：

电话：(020)28267800 22644385

邮箱：serial@embedcontrol.com

编程器：

电话：(020)22644371

邮箱：programmer@embedtools.com

分析仪器：

电话：(020)22644375

邮箱：tools@embedtools.com

ARM 嵌入式系统：

电话：(020) 22644383 22644384

邮箱：NXPARM@zlgmcu.com

楼宇自动化：

电话：(020)22644376 22644389 28267806

邮箱：mjs.support@ecardsys.com

mifare.support@zlgmcu.com

销售：

电话：(020)22644249 22644399 22644372 22644261 28872524

28872342 28872349 28872569 28872573 38601786

维修：

电话：(020)22644245

目 录

1. TinyM0 T12M 开发工具

TinyM0 T12 和 TinyM0 T12M 是广州致远电子有限公司为企业用户和高校师生设计的两款开发工具，核心微控制器采用 NXP 公司最新推出的 LPC1200 系列 Cortex-M0 内核芯片。



LPC1200 微控制介绍：

LPC1200 系列微控制器采用了 ARM 公司最新发布的 Cortex-M0 内核，工作频率高达 45MHz，性能卓越、简单易用、功耗极低，更重要的是，它能显著降低所有 8 位/16 位应用的代码长度，其价值和易用性比现有的 8 位/16 位微控制器更胜一筹，为追求 ARM 架构的 8/16 位用户提供了一种全新的 32 位解决方案。

TinyM0 T12 和 TinyM0 T12M 开发板实物如图 1.1、图 1.2 所示，由 TKScope CK100 仿真器和核心板两部分组成，中间通过邮票孔连接。开发板可以整体使用，也可断开独立使用，应用简单灵活。

注：TinyM0 T12M 开发板和 TinyM0 T12 开发板核心 MCU 均为 LPC1227，只是 MCU 封装不同，前者为 LQFP64 封装，后者为 LQFP48 封装。LQFP64 封装比 LQFP48 封装多一个 P2 口，P2 口具有 16 个 GPIO 引脚。本文将以 TinyM0 T12M 为例进行深入介绍，TinyM0 T12 用法与其相同。

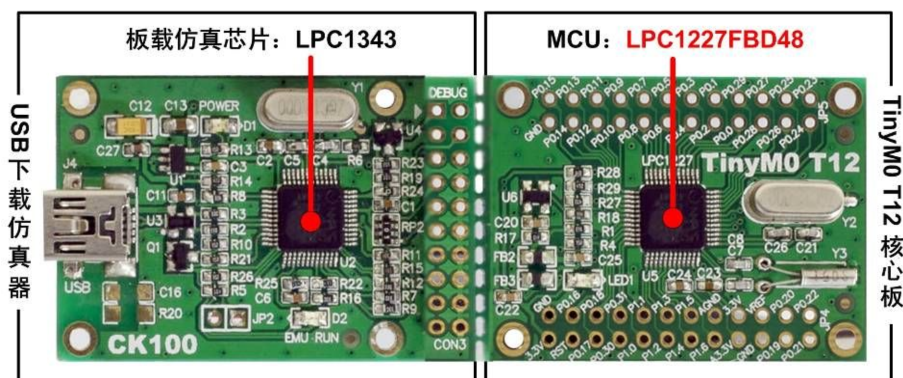


图 1.1　TinyM0 T12 开发板实物图

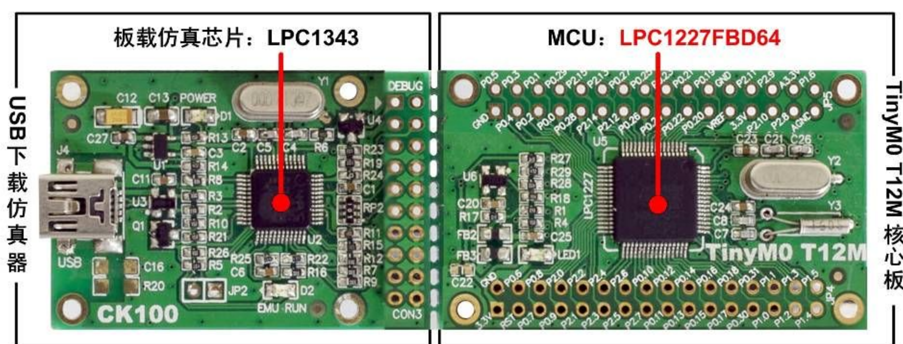


图 1.2　TinyM0 T12M 开发工具实物图

TinyM0 T12M 开发工具特点：

1. 板载仿真器

TinyM0 T12M 板载 TKScope CK100 仿真器，支持目前市场上的 IAR、TKStudio 等主流集成开发环境。

2. 配套核心板

TinyM0 T12M 核心板电路为 LQFP64 引脚封装 LPC122x 芯片的最小系统，硬件支持 2.54mm 间距的标准排针。TinyM0 T12M 核心板断开后，可配套用户自行设计的底板，直接进行产品开发。

3. 支持多款芯片

TinyM0 T12M 核心板全面支持 NXP LPC1200 系列 LQFP64 引脚封装的芯片，用户可以根据设计要求随时更换核心控制器。

4. 配套资料

TinyM0 T12M 开发板配套提供详细的开发指南以及丰富的实验例程，是工程师学习、开发 Cortex-M0 微控制器的不二之选。

2. TKStudio IDE集成开发环境

本章将介绍如何使用 TKStudio 集成开发环境进行工程的建立、编译链接和调试等操作，帮助用户快速掌握 Cortex-M0 芯片的开发环境。

2.1　TKStudio 集成开发环境

TKStudio 集成开发环境(又称 TKStudio IDE)是广州致远电子有限公司开发的一个微处理器软件开发平台，是一款具有强大内置编辑器的多内核编译调试环境，支持 8051、ARM、AVR 等多种微控制器，可以完成从工程建立和管理，编译，链接，目标代码的生成，到软件仿真，硬件仿真(挂接 TKS 系列仿真器等硬件)等完整的开发流程。TKStudio IDE 主界面如图 2.1 所示。

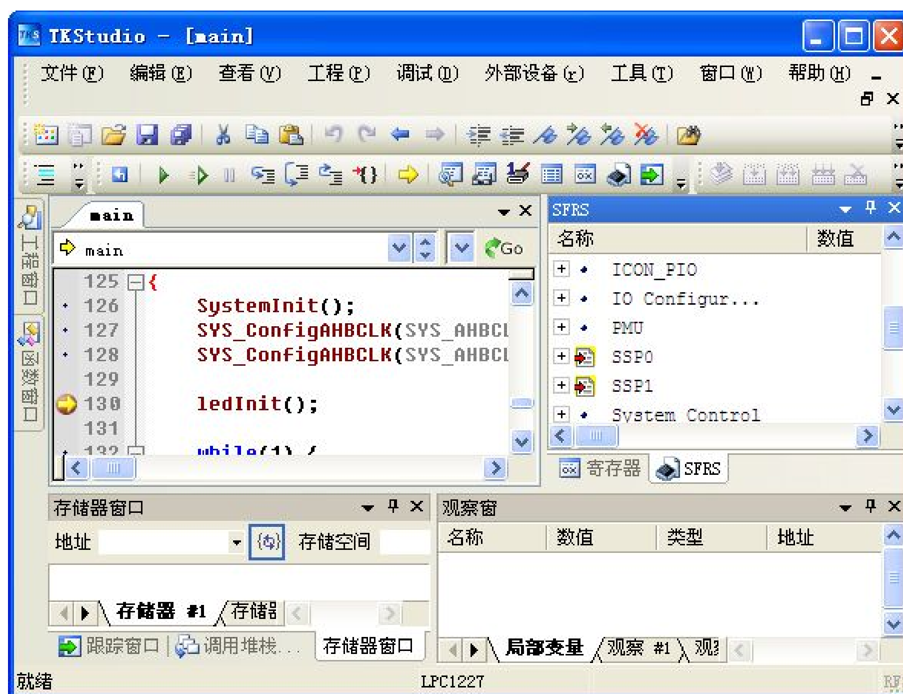


图 2.1　TKStudio 主界面

TKStudio 集成开发环境包括工程管理器、代码编辑器、编译工具链、源码级调试器和外部工具等。

2.1.1 工程管理器

TKStudio 工程管理器可以管理工程中用到的所有源文件、库文件和其它输入文件。在工程窗口中分为工程、文件组、源文件三级结构，并且根据文件类型显示为不同的图标，直观而又醒目，更方便用户管理工程。TKStudio 工程窗口如图 2.2 所示。

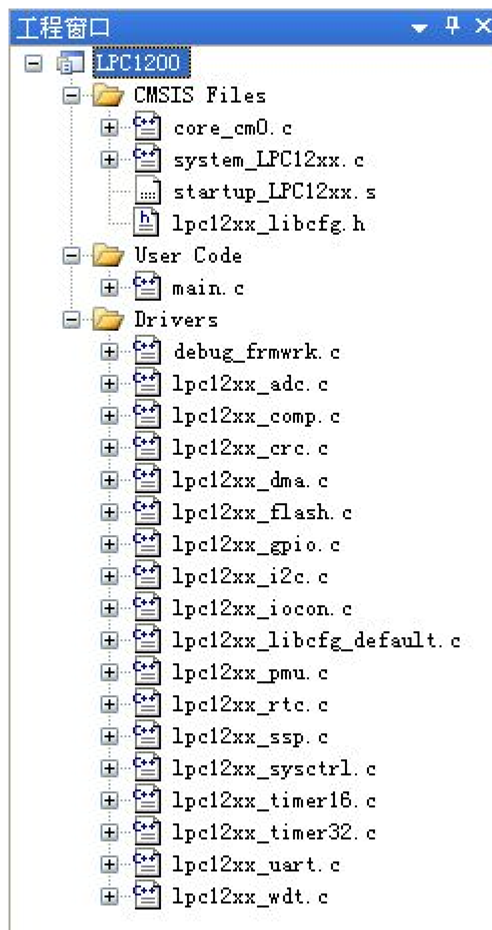


图 2.2 工程窗口

2.1.2 代码编辑器

TKStudio 代码编辑器以最大化用户编码体验为目标，功能强大，能非常有效地提高编码效率，并提高整体开发效率。TKStudio 代码编辑器功能特点总结如图 2.3 所示。



图 2.3 TKStudio 代码编辑器功能特点

2.1.3 编译工具链

TKStudio 具备丰富的编译器选项配置信息，对各种工具链的编译、链接、调试提供了灵活的配置参数，对 MCS-51、ARM、AVR 等芯片都提供了支持，主要有 C51、SDCC 51、

GCC ARM、ADS ARM、Realview MDK、GCC AVR 等编译工具链。

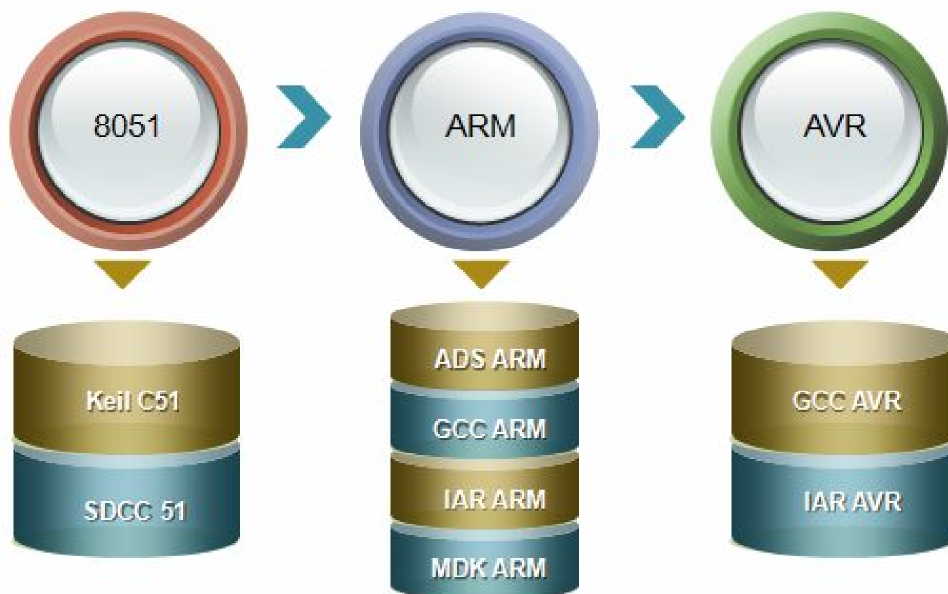


图 2.4　编译工具链

LPC1200 系列微控制器开发需要用到的编译工具链是 Realview MDK 4.x，如果用户在安装 TKStudio 前已经安装了该编译器，安装程序会自动将编译器的路径设置到 TKStudio 中。但如果用户之前没有安装 Realview MDK 编译器，则需在 TKStudio 安装过程中根据提示到相关网站下载安装。

在安装好 Realview MDK 编译器后，用户需要将该编译器的路径设置到 TKStudio 中。打开 TKStudio 集成开发环境，如图 2.5 所示，选择【工程】→【编译工具设置】菜单，将弹出如图 2.6 所示对话窗口。

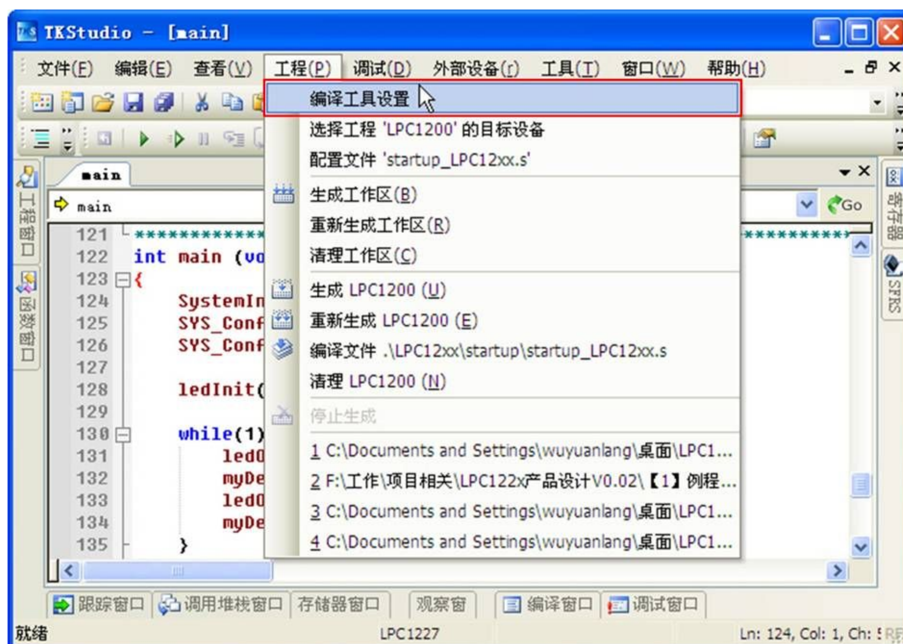


图 2.5　打开编译工具设置

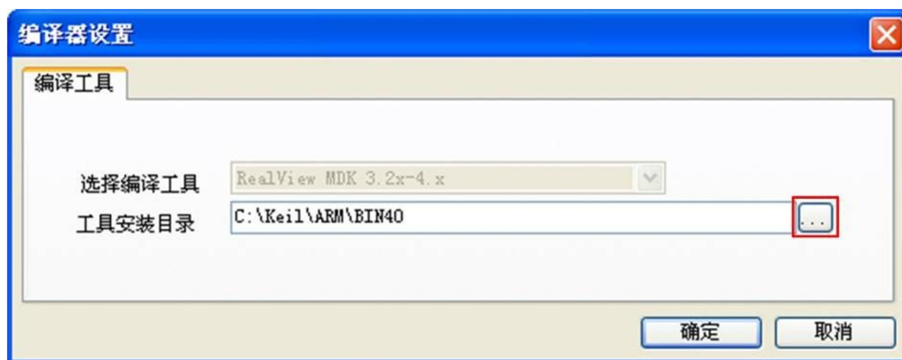


图 2.6　编译器设置对话框

点击按钮[...]，选择 Realview MDK 所在路径，如图 2.7 所示，注意需要定位到【KEIL】→【ARM】→【BIN40】目录，然后点击【确定】，即完成了编译工具的设置。



图 2.7　选择编译工具路径

2.1.4 源码级调试器

TKStudio 调试器与广州致远电子推出的 K 系列仿真器完美结合，支持 8051、ARM、AVR 等主流芯片的系统级调试，配备大量调试窗口、性能分析器、数据/代码覆盖分析器、运行轨迹回溯工具，内置了完备的软件仿真系统。能帮助您快速定位无用代码、无用变量，更好地优化完善整个系统。



图 2.8　调试状态

2.1.5 外部工具

主界面上的【工具】菜单下包含了一些实用的外部工具，包括烧写工具、调试工具、文件工具、计算器等等，可以根据自己的需要灵活使用这些工具。

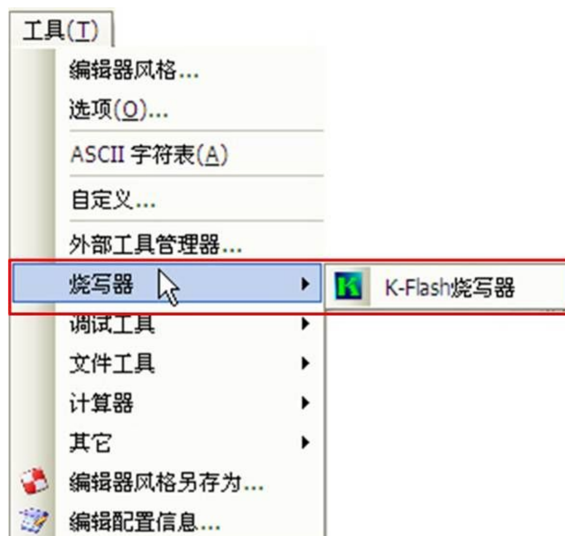


图 2.9　外部工具

本文第 4 节与第 5 节将对开发中常用的两款软件：K-Flash 烧写器和 TKS_COM 串口调试助手，进行详细的介绍。

2.2　TKStudio 安装

2.2.1 欢迎界面

双击 TKStudio IDE(M0 专用版)安装程序，弹出欢迎界面如图 2.10 所示，点击【下一步】继续。

注：TKStudio IDE(M0 专用版)下载地址：http://www.zlgmcu.com/NXP/LPC1000/ds/TKSetup_M0.exe

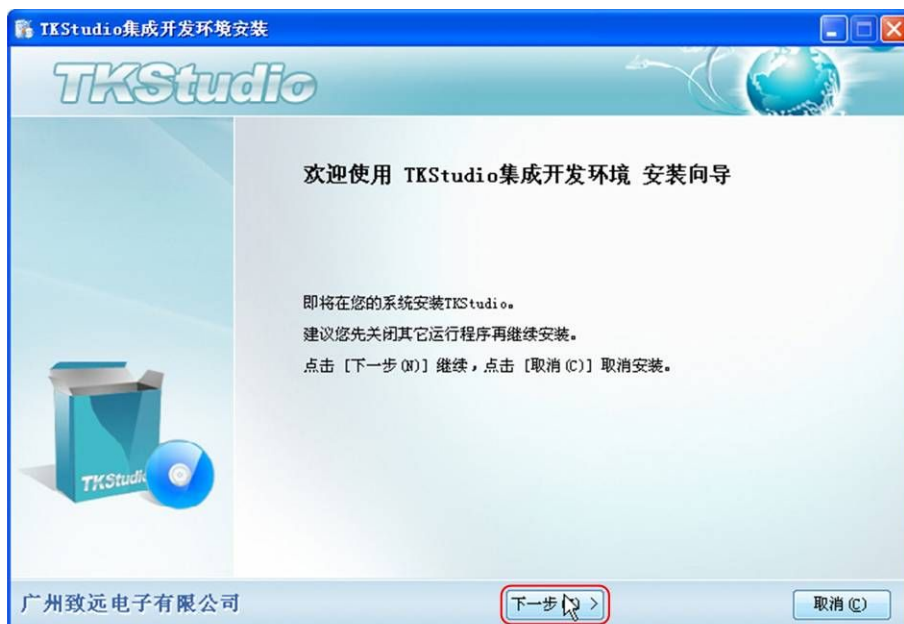


图 2.10　欢迎界面

2.2.2 阅读许可证协议

许可证协议描述当前版本许可证协议授于用户的权利和相关规定，如图 2.11 所示。仔细阅读后，选择【我接受“许可证协议”中的条款】，点击【下一步】继续。

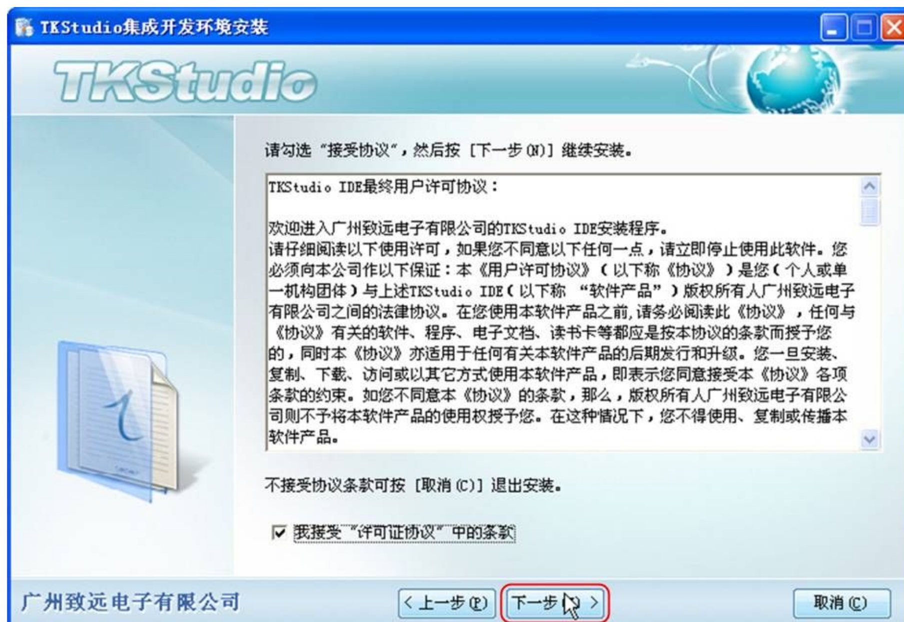


图 2.11　许可证协议

2.2.3 选择安装路径

选择安装路径描述 TKStudio IDE 的安装路径，如图 2.12 所示。默认路径为 C:\Program Files\Zhiyuan Electronic\TKStudio，您也可以通过点击【浏览】或直接修改目标文件夹的位置，点击【下一步】继续。



图 2.12　安装路径

2.2.4 设置开始菜单

设置开始菜单描述了在开始菜单中要显示的 TKStudio 的名字，用户可以根据自己的需要进行修改，如图 2.13 所示。点击【下一步】开始安装。

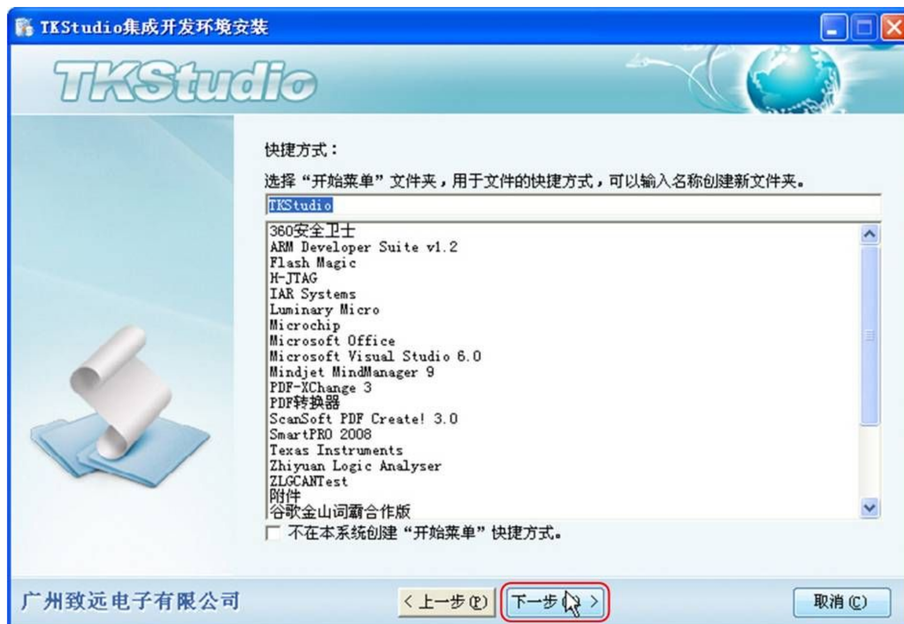


图 2.13　快捷方式

2.2.5 正在安装

正在安装页面显示当前安装进度，并播放 TKStudio 的特性，此过程由程序自动控制，如图 2.14 所示。



图 2.14 正在安装

2.2.6 完成程序安装

完成程序安装页面提示用户已安装完程序，点击【完成】按钮即完成了 TKStudio IDE 的安装，如图 2.15 所示。

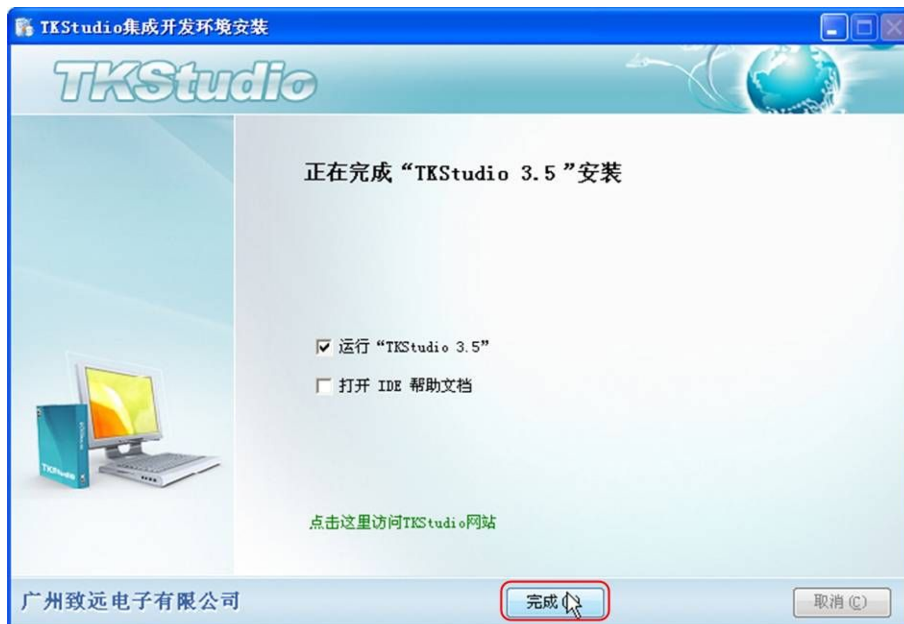


图 2.15 完成安装

2.3 工程的管理

2.3.1 新建工程

点击 Windows 操作系统的【开始】→【程序】→【TKStudio】菜单下的【TKStudio】开启运行 TKStudio，如图 2.16 所示，或者双击桌面上的 TKStudio 快捷方式也可以打开运行。



图 2.16　运行 TKStudio

运行 TKStudio 后，在如图 2.1 所示主界面上选择菜单【文件】→【新建】，弹出对话框后默认为【文件】，点击选择【工程】→【Project Wizard】，如图 2.17 所示。

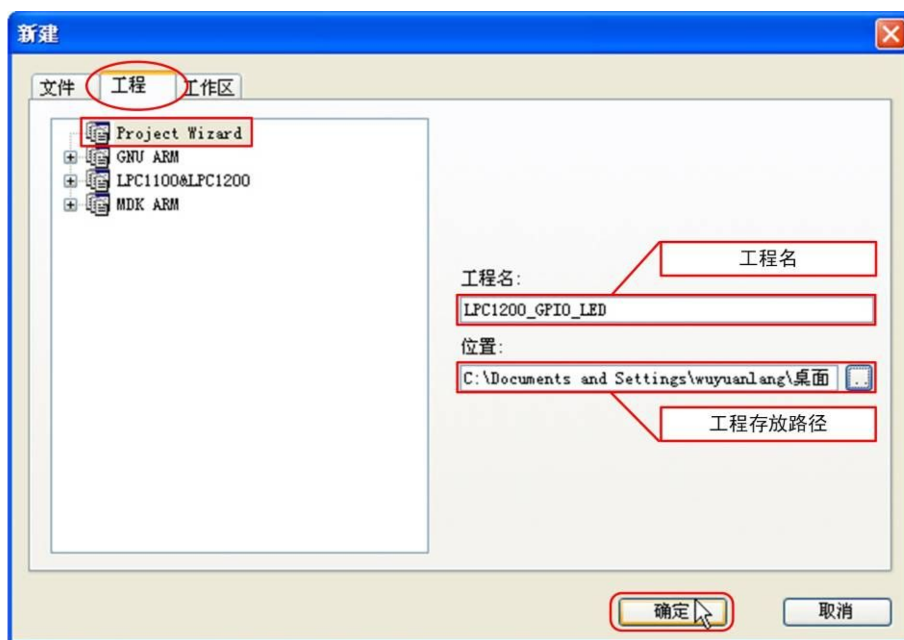


图 2.17　新建工程

在【位置】项中选择工程存放路径，在【工程名】项中输入工程名称，并单击【确定】，将弹出【选择目标 CPU】对话框，如图 2.18 所示。



图 2.18　选择目标 CPU

在【选择内核】项中选择器件类型，在【选择编译工具】项选择编译工具链，可在【搜索栏】项中直接搜索芯片名称，或在【器件库】项中选择具体芯片，然后单击【确定】即可建立相应工程，工程文件名后缀为【.xmp】。

2.3.2 从工程模板新建工程

针对 LPC1200 系列微控制器，我们定义了 1 个专用工程模板，包含设置信息由 Flash 起始地址 0x00000000、片内 RAM 起始地址 0x10000000、编译链接选项及编译优化级别等；模板中包含了 LPC1200 系列微控制器的启动文件(如：startup_LPC12xx.s)、驱动库文件等。

建议用户开发基于 LPC1200 工程模板，从模板新建工程具体操作如下：

点击 Windows 操作系统的【开始】→【程序】→【TKStudio】菜单下的【TKStudio】开启运行 TKStudio，如图 2.19 所示，或者双击桌面上的 TKStudio 快捷方式也可以打开运行。



图 2.19　运行 TKStudio

运行 TKStudio 后，在如图 2.1 所示主界面上选择菜单【文件】→【新建】，弹出对话框后默认为【文件】，点击选择【工程】，选择【Cortex-M0_Execute_for_LPC1200】工程模板，在【位置】项中选择工程存放路径，在【工程名】项中输入工程名称，并单击【确定】，如图 2.20 所示。

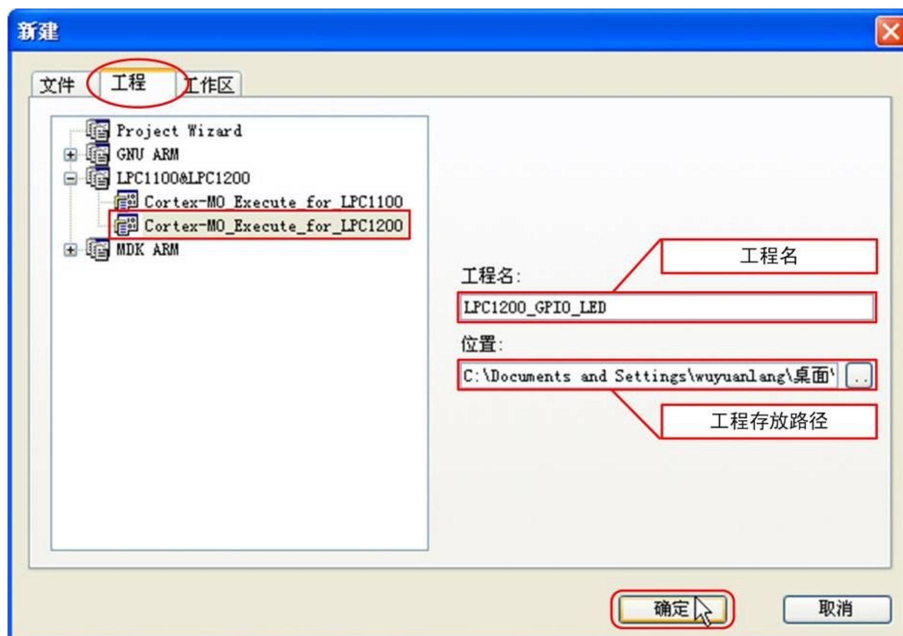


图 2.20　从工程模板新建工程

在如图 2.1 所示主界面上选择菜单【工程】→【选择工程‘LPC1200_GPIO_LED’的目标设备】，如图 2.21 所示。弹出选择目标 CPU 对话框如图 2.22 所示。

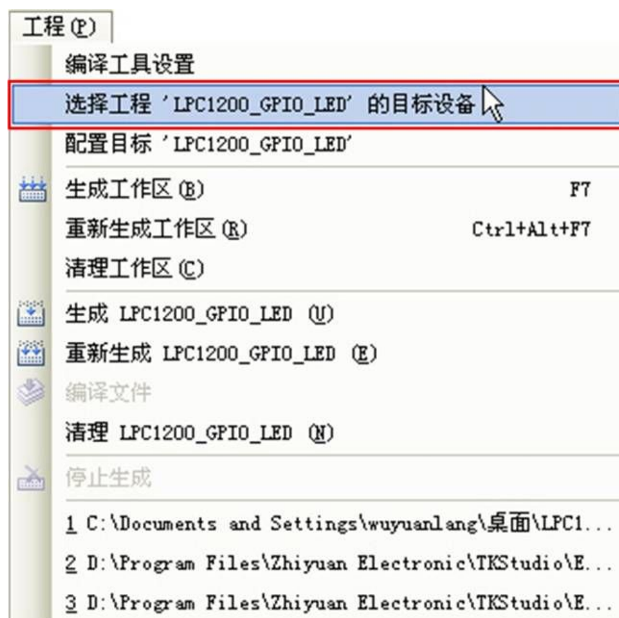


图 2.21　选择工程的目标设备



图 2.22　选择目标 CPU

可在【搜索栏】项中直接搜索芯片名称，或在【器件库】项中选择具体芯片，然后单击【确定】即完成选择目标 CPU。

LPC1200 专用工程模板说明如下：

Cortex-M0_Execut_for_LPC1200：无操作系统，所有 C 代码均编译成 Thumb 指令。

用户选择相应的工程模板建立工程，如图 2.23 所示为使用【Cortex-M0_Execute_for_LPC1200】工程模板建立的一个工程。工程有一个生成目标：DebugInFlash。工程模板已经将相应的编译参数设置完成，用户直接使用即可。

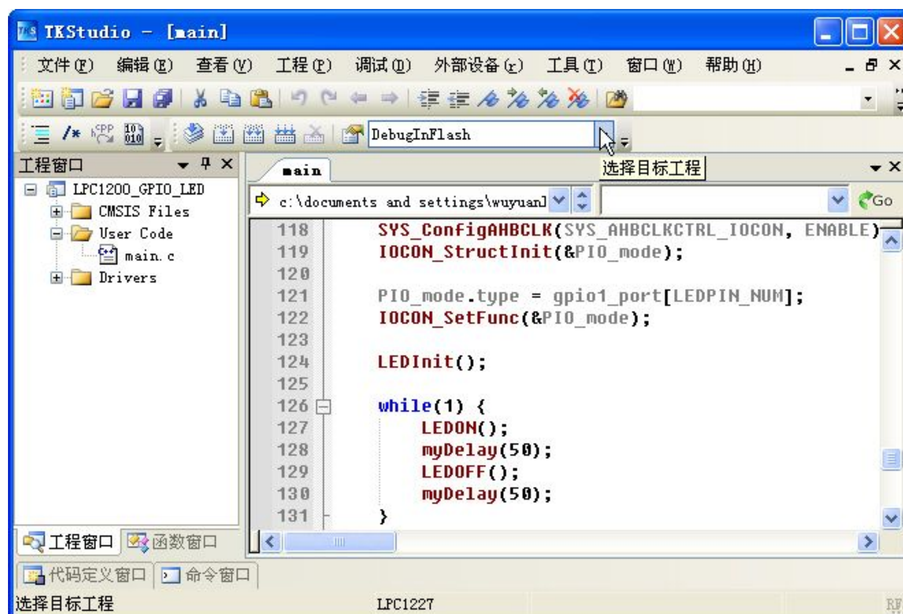


图 2.23　采用 LPC1200 工程模板建立的工程

2.3.3 建立文件

建立一个文本文件，以便输入用户程序。如图 2.24 所示，点击【新文件】或点击菜单

栏的【文件】→【新文件】，然后在新建的文件中编写程序，如图 2.25 所示，单击【保存】按钮将文件存盘，或选择【文件】→【保存】，弹出对话框如图 2.26 所示，在【文件名】项中输入文件全名，如 main.c，单击【保存】即可。

注意:

1、 保存的文件名需要添加文件类型，如 c 文件则命名为 main.c，汇编文件命名为 main.asm;

2、 请将文件保存到相应工程目录下，以便于管理和查找。



图 2.24 新文件按钮



图 2.25 保存按钮



图 2.26 保存新建文件

也可以通过在菜单栏选择【文件】→【新建】按钮或按快捷键【Crtl+N】弹出【新建】对话框建立文件，如图 2.27 所示。

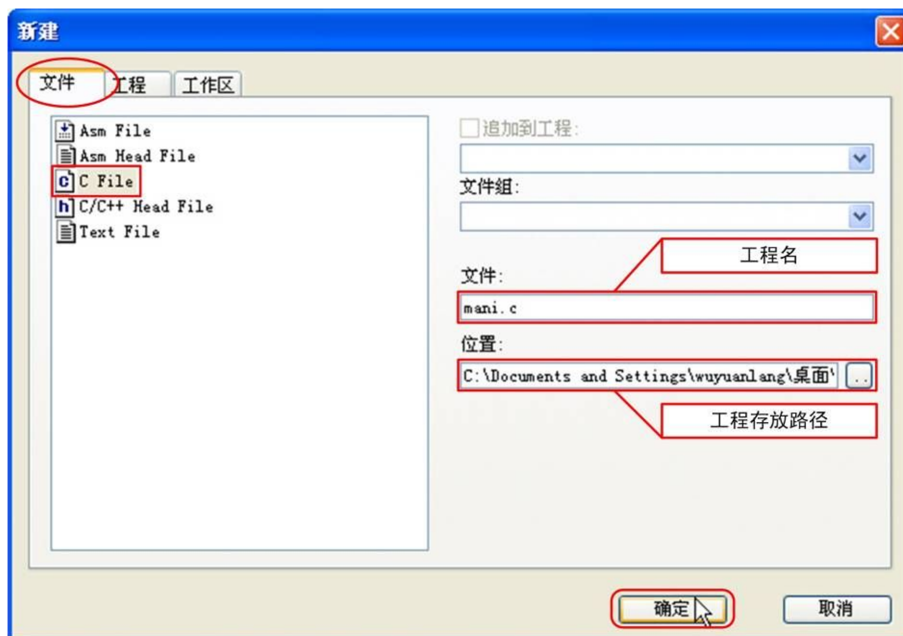


图 2.27　新建文件

2.3.4 添加文件到工程

如图 2.28 所示，在工程窗口中，选择对应的文件组，单击右键，在弹出菜单中选择【追加文件到文件组】即可弹出【打开】对话框如图 2.29 所示，选择要添加的文件，双击或单击【打开】即完成文件添加操作。

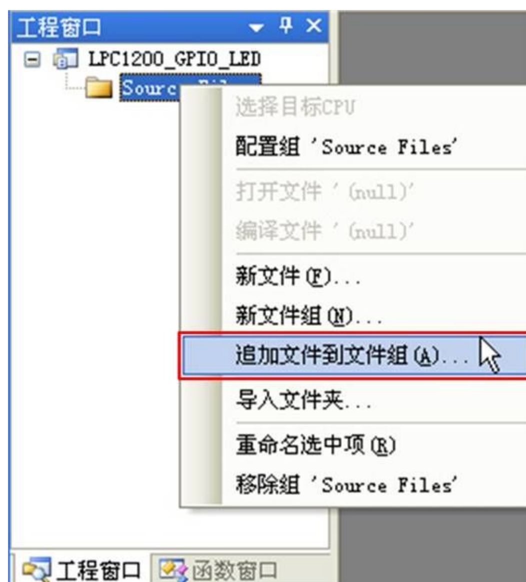


图 2.28　在工程窗口中添加源文件

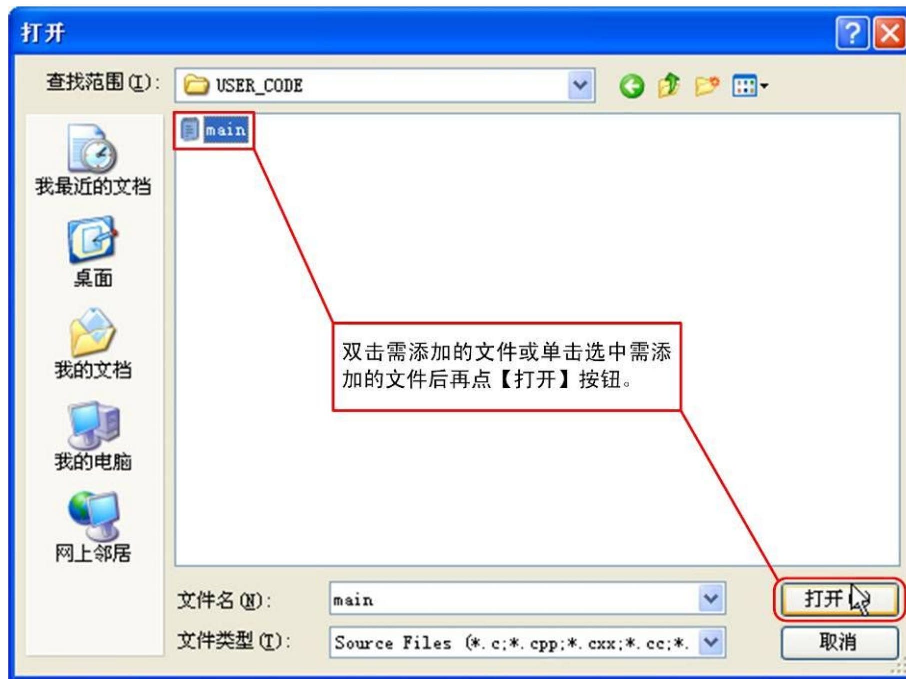


图 2.29　打开需添加的文件

2.3.5 编译链接工程

如图 2.30 所示为编译链接工程所使用的工具条，通过这些按钮，可以快速地进行工程设置、编译链接等(在不同的菜单项上可以分别找到对应的菜单命令)。



图 2.30　编译链接工具条

表 2.1　编译链接工具条说明

按钮	说明
	编译选定文件，编译结果将显示在编译窗口，如果出现错误或警告，双击错误信息或警告信息光标就会定位到出错的源代码行
	生成工程，编译修改过的文件并生成应用
	重新生成工程，重新编译当前工程所有的文件并生成应用
	生成所有工程，重新编译所有的文件并生成应用
	停止生成，停止正在进行的编译或构建过程
	工程配置，配置目标工程相关参数
DebugInFlash	当前的目标工程

点击按钮，将弹出一个配置目标工程对话框如图 2.31 所示，在该对话框中可以对目

标工程进行地址设置、输出文件设置和编译项设置等等。

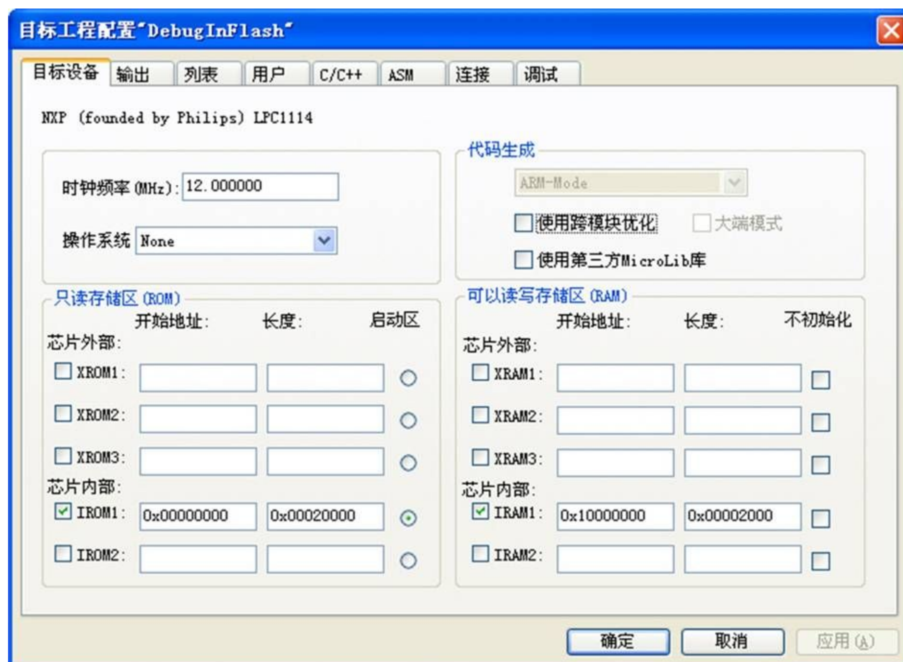


图 2.31　工程配置窗口

2.3.6 打开工程

如图 2.32 所示，在主界面菜单栏选择【文件】→【打开工程或工作区】即可弹出打开对话框如图 2.33 所示，找到相应的工程文件(*.xmp)，双击或单击【打开】按钮即可。如果需要打开最近的工程，在菜单栏【工程】项下面的最近打开工程中选择即可，如图 2.34 所示。

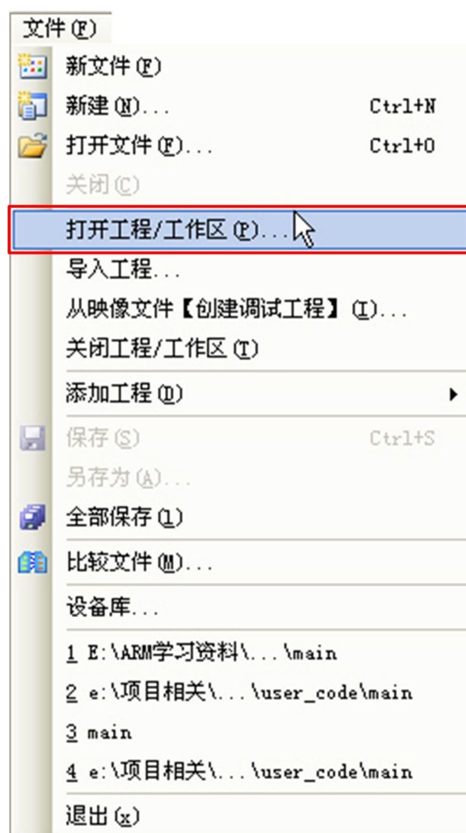


图 2.32　打开工程

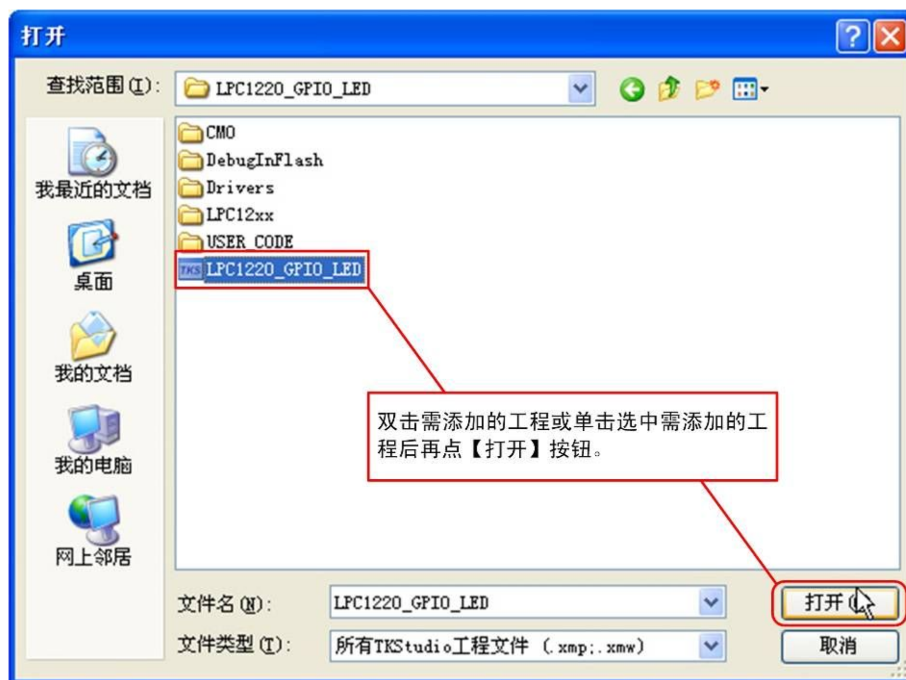


图 2.33　需打开工程源文件

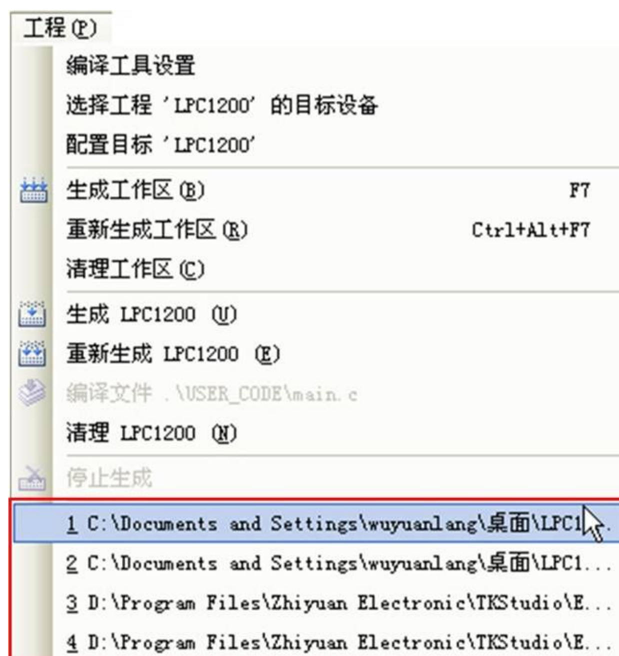


图 2.34　最近打开工程

2.3.7 导入工程

导入工程即导入一个非 TKStudio 工程，TKStudio 目前支持 uV2、uV3、ADS、SDCC 工程的导入。如图 2.35 所示，选择【文件】→【导入工程】菜单，即弹出如图 2.36 所示对话框。

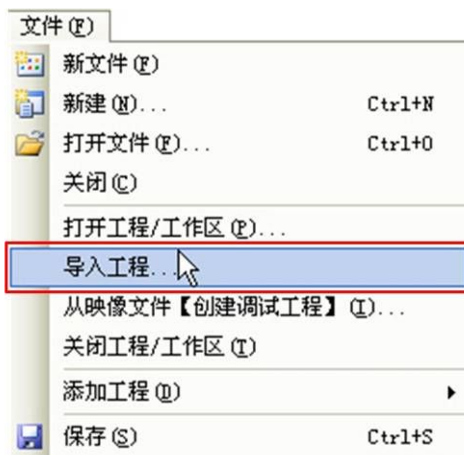


图 2.35　导入工程

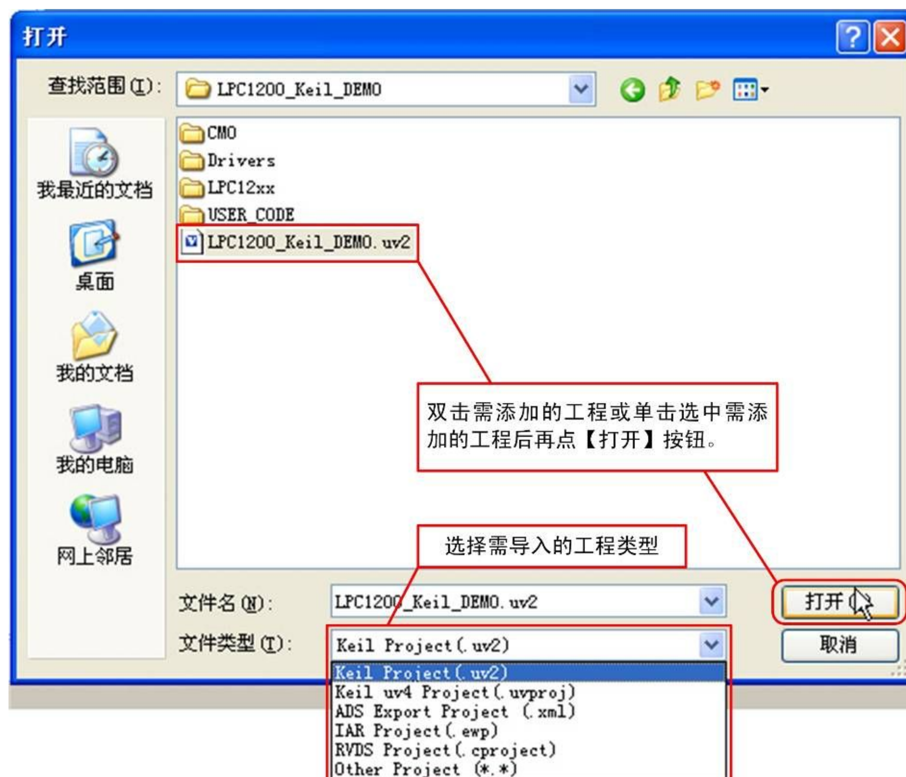


图 2.36　导入工程源文件

在文件类型下拉列表框中选择要导入的工程类型，选择工程，双击或单击【打开】按钮，将弹出【选择目标 CPU】的对话框，如图 2.18 所示，设置好相应的配置信息后点击【确定】，弹出另存工程对话框如图 2.37 所示，选择一个保存工程路径(此处一定要把工程文件保存到原工程的同目录下)，点击【保存】即可把一个非 TKStudio 工程导入到 TKStudio 中进行编辑或编译等操作了。

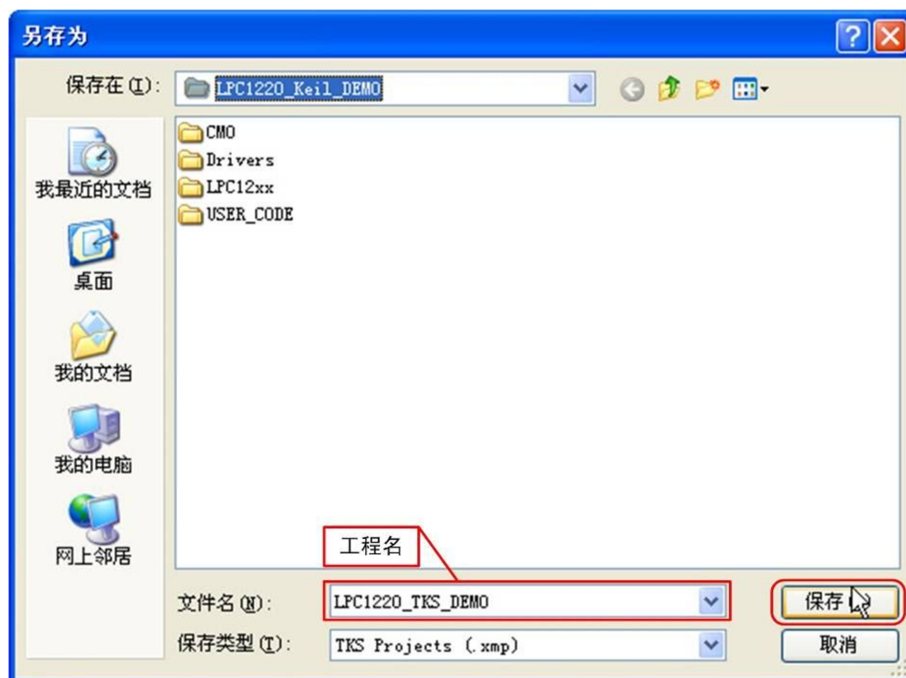


图 2.37　导入工程另存

2.4 工程的调试

2.4.1 选择调试方式

调试主要有两种方式：软件仿真和硬件调试。在 TKStudio 中，通过工程配置对话框可选择不同的调试方式，并设置调试参数。不同的工具链下会有不同工程配置对话框。如选择 MDK 编译工具时，调试设置界面如图 2.38 所示。

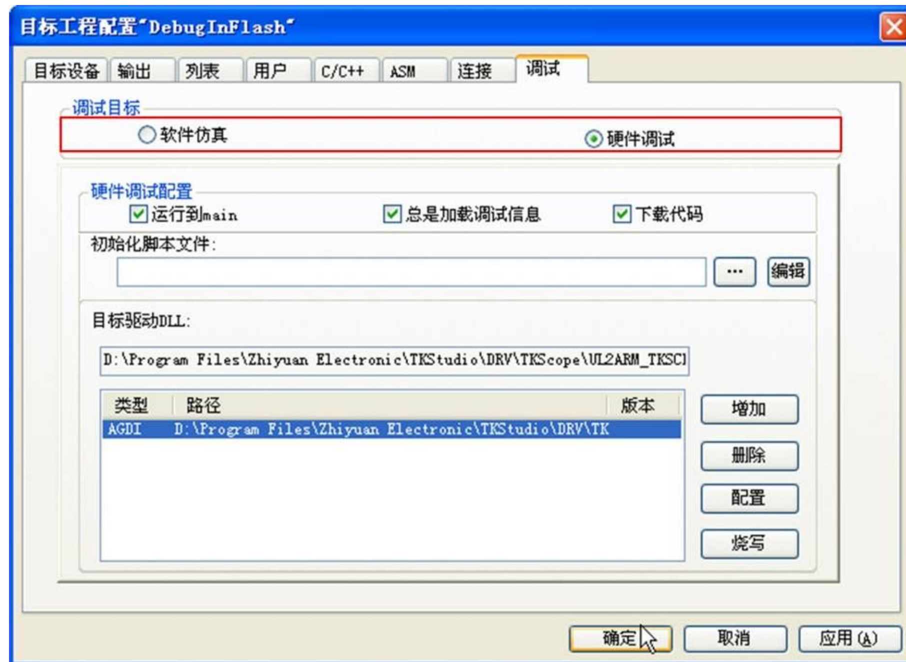


图 2.38　选择调试方式

当需要配置软件仿真或硬件调试选项时，可以点击相应的单选框切换到相应的配置界面。

2.4.2 调试工具条

当工程编译连接通过后，单击【启动/停止调试】按钮，即可启动 TKStudio 进行调试。TKStudio 启动调试和断点操作工具条如图 2. 39 所示。



图 2.39　启动调试和断点操作工具条

表 2.2　启动调试和断点操作工具条说明

按钮	说明
	启动/停止调试
	插入/移除断点
	启用/禁止断点
	禁止所有断点

续上表

按钮	说明
	清除所有断点

TKStudio 调试工具条如图 2.40 所示，工具条说明见表 2.3。



图 2.40　调试工具条

表 2.3　调试工具条说明

按钮	说明
	复位
	运行
	忽略断点运行
	停止运行
	步进-单步运行，进入调用函数
	步越-单步运行，调用函数作为一条语句执行
	步出，执行完当前被调用的函数，停止在函数调用的下一条语句
	运行到光标所在行，运行程序直到当前光标所在行时停止
	显示下一个即将执行的指令行
	显示或隐藏反汇编窗口
	显示或隐藏观察窗口
	显示或隐藏串行监视窗口
	显示或隐藏存储器窗口
	显示或隐藏寄存器窗口
	显示或隐藏特殊功能寄存器(SFR)窗口
	显示或隐藏跟踪窗口

2.5 LPC1200 代码安全

代码读保护是允许用户在系统中通过使能不同的安全级别来限制对片内 Flash 的访问和 ISP 的使用的一种机制。需要时，可通过在 Flash 地址单元 0x0000 02FC 编程特定的格式来调用 CRP。IAP 命令不受代码读保护的影响。代码安全选项详细说明如表 2.4 所示。

注：CRP 所作出的任何改变只有在器件经过一个电源周期之后才会生效。

表 2.4 代码读保护选项

名称	在 0x000002FC 处编程的格式	描述
NO_ISP	0x4E697370	阻止对 PIO0_12 管脚进行采样而进入 ISP 模式。PIO0_12 管脚作其他用途
CRP1	0x12345678	禁止通过 JTAG 引脚访问芯片。该模式允许在下列条件约束下使用下列 ISP 命令来进行局部的 Flash 更新： ● 写 RAM 命令不能访问在 0x10000300 以下的 RAM； ● 将 RAM 内容复制到 Flash 命令不能写扇区 0； ● 只有在选择擦除所有扇区时，擦除命令才能擦除扇区 0； ● 比较命令禁能； ● 禁能读取存储器命令。 当需要 CRP 且要更新 Flash 字段时可使用该模式，但不能擦除所有扇区。由于在 Flash 部分更新的情况下比较命令禁能，因此次级装载程序应执行校验和机制来验证 Flash 的完整性
CRP2	0x87654321	禁止通过 JTAG 引脚访问芯片。下列的 ISP 命令被禁能： ● 读存储器； ● 写 RAM； ● 运行； ● 将 RAM 内容复制到 Flash； ● 比较。 使能 CRP2 时，ISP 擦除命令仅允许擦除所有用户扇区的内容
CRP3	0x43218765	禁止通过 JTAG 引脚访问芯片。如果 Flash 扇区 0 中存在有效用户代码，则禁止通过拉低 PIO0_12 来进入 ISP。 该模式有效的禁止了通过 PIO0_12 脚来强行进入 ISP 的行为。用户的应用程序可决定是调用 IAP 来进行 Flash 更新还是通过 UART0 重新调用 ISP 命令来进行 Flash 更新 **注：如果选择了 CRP3，不能通过 ISP 擦除芯片，请慎重使用该模式**

1. 加密操作

双击工程窗口中 startup_LPC12xx.s 打开启动文件，如图 2.41 所示。【CRP_Key】行中 DCD 后定义的默认值为 0xFFFF FFFF，即不加密，用户只需将该值修改为表 2.4 列出的 CRP 值，即可实现不同等级的代码加密。

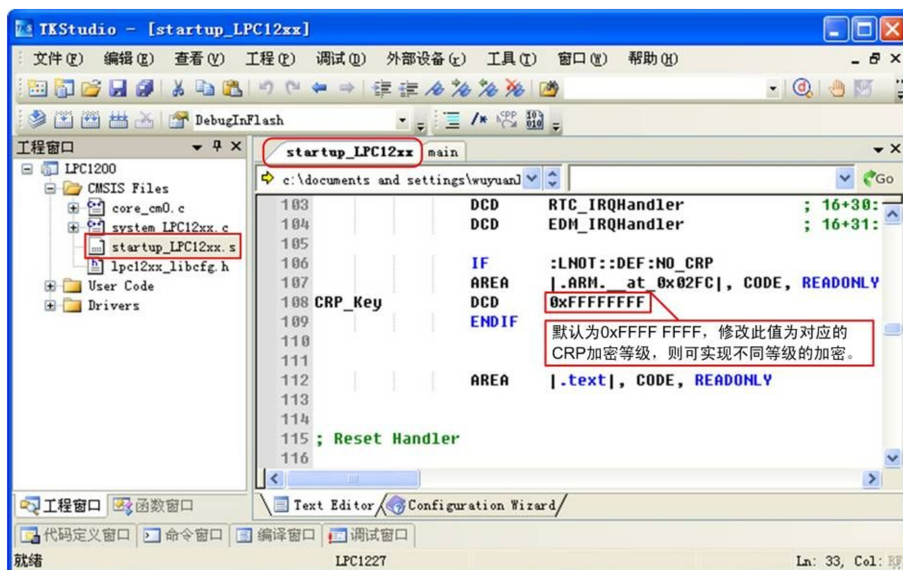


图 2.41 打开启动文件

2. 查看加密效果

用户可以通过查看工程编译后生成的 BIN 文件确认 CRP 值是否写入到 Flash 地址单元 0x0000 02FC。

打开工程配置窗口，选择【输出】项，如图 2.42 所示，在输出文件格式下拉菜单中选择【BIN 文件】，并勾选【产生浏览信息】，然后点击【确定】按钮退出，再重新编译工程。

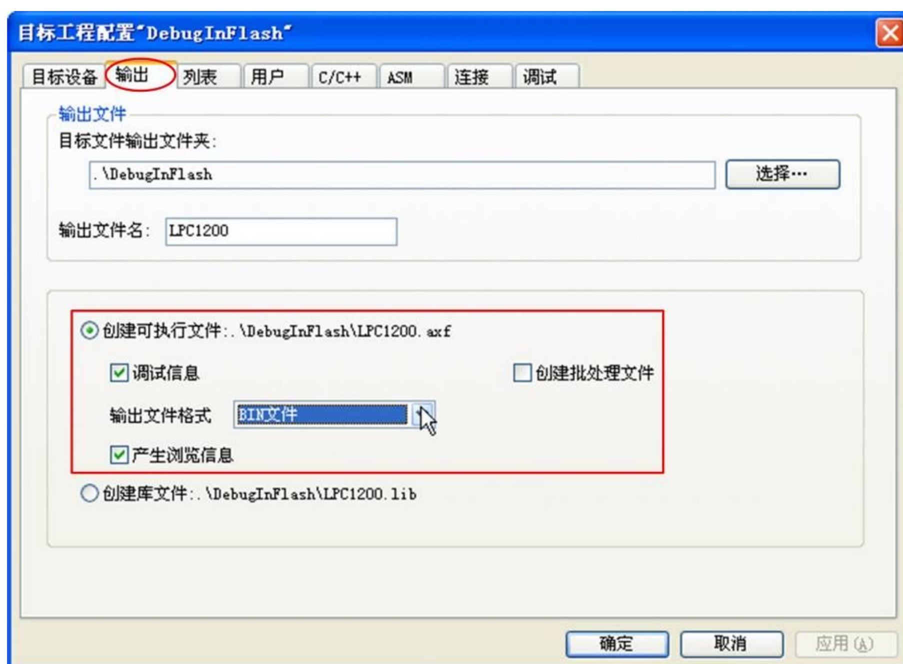


图 2.42 设置生成 BIN 文件

重新编译工程后，会生成一个【LPC1200.bin】文件(在工程目录下的 DebugInFlash 文件夹中)，找到该文件并用 TKStudio 打开，点击图 2.43 中【二进制显示】按钮，查看地址 0x0000 02FC 处的值，即可确认设定的 CRP 值是否正确写入。

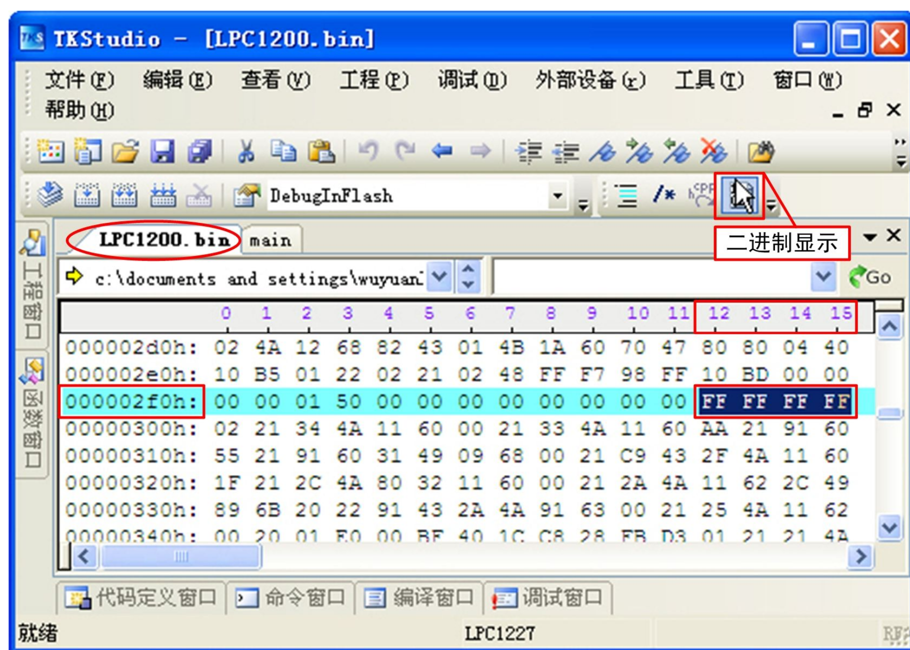


图 2.43　查看 BIN 文件

3. TKScope CK100 仿真器

3.1 功能简介

TKScope CK100 仿真器是广州致远电子有限公司 2009 年隆重推出的一款高性能 ARM 专用仿真器，是支持 Cortex-M0 内核的仿真器。随之不断升级，将陆续推出支持 ARM7、ARM9、Cortex-M1、Cortex-M3、XSCALE、ARM10、ARM11、Cortex-R4 和 Cortex-A8 等内核的全系列仿真。

TKScope CK100 仿真器与多种主流 IDE 环境无缝嵌接，支持 IAR 和 TKStudio 等开发环境，并具备其高级调试功能，保证用户的开发平台始终如一。



目前，TKScope CK100 仿真器支持的 IDE 环境如下：

- **TKStudio**：致远公司，中/英文界面，多内核编译/调试环境，强大内置编辑器；
- **IAR**：IAR 公司，英文界面，多内核编译/调试环境。

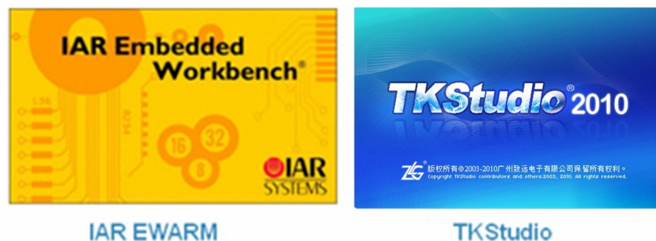


图 3.1 TKScope CK100 仿真器支持的 IDE 环境

NXP 公司的 Cortex-M3 内核控制器 LPC1343 内部集成 HID 和 MSC 驱动固件，TKScope CK100 仿真器正是利用此特性，选用 LPC1343 作为核心控制器，采用 USB 高速通讯接口，快速下载用户程序代码。同时，USB 接口从 PC 机取电，省去沉重的电源适配器，外形美观小巧，携带方便。

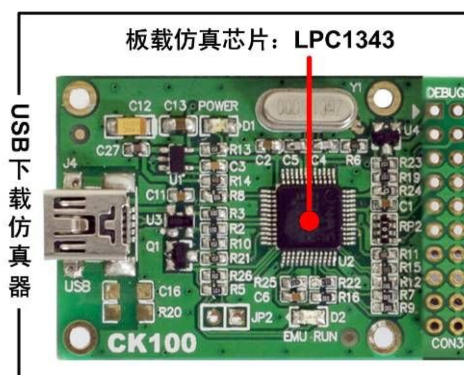


图 3.2 TKScope CK100 仿真器

3.2 性能特点

TKScope CK100 仿真器是 TKScope SuperARM ICE CK 系列的仿真产品，只专注于 ARM 内核的仿真，是 TKScope 仿真器的 ARM 单机版。

TKScope 简介：

TKScope 嵌入式综合仿真开发平台，支持 8051、ARM、DSP、AVR、C166、MX 等内核全系列仿真，是一款全新概念的高级专业通用型仿真开发平台；与多种主流 IDE 环境无缝嵌接，支持 TKStudio、zlglogic、Keil、ADS、IAR、RealView、AVR 和 CCS 等开发环境；同时，内嵌专业的逻辑分析仪，zlglogic 软件全面支持。

TKScope CK100 仿真器主要特性：

- 支持 Cortex-M0 内核；
- 支持 Thumb 模式，支持 SWD 仿真模式；
- 与主流 IDE 环境无缝嵌接，如 IAR、TKStudio 等；
- 支持片内 Flash 在线编程/调试，提供每种芯片对应的 Flash 编程算法文件；
- 支持最多 2 个硬件断点；
- JTAG 时钟最高可达 1MHz，且 JTAG 时钟在允许范围内可任意调整；
- 基于芯片的设计理念，为每款芯片提供完善的初始化文件；
- 带有硬件自检功能，方便检测排除硬件故障。

注：广州致远电子有限公司有权在任何情况下，对 TKScope CK100 仿真器进行功能上的限制及裁剪。

3.3 安装驱动

TKScope CK100 仿真器通过 mini USB 接口与 PC 机连接，支持热插拔。开发板上电后红色电源指示灯【POWER】将点亮，绿色仿真器指示灯【EMU RUN】闪烁一次后熄灭。

开发板第一次上电时，PC 机屏幕右下脚将弹出【发现新硬件】的提示信息，如图 3.3 所示，在如图 3.4 窗口选择【从列表或指定位置安装(高级)】选项，然后单击【下一步】，此时系统会弹出如图 3.5 所示的对话框。



图 3.3 发现新硬件

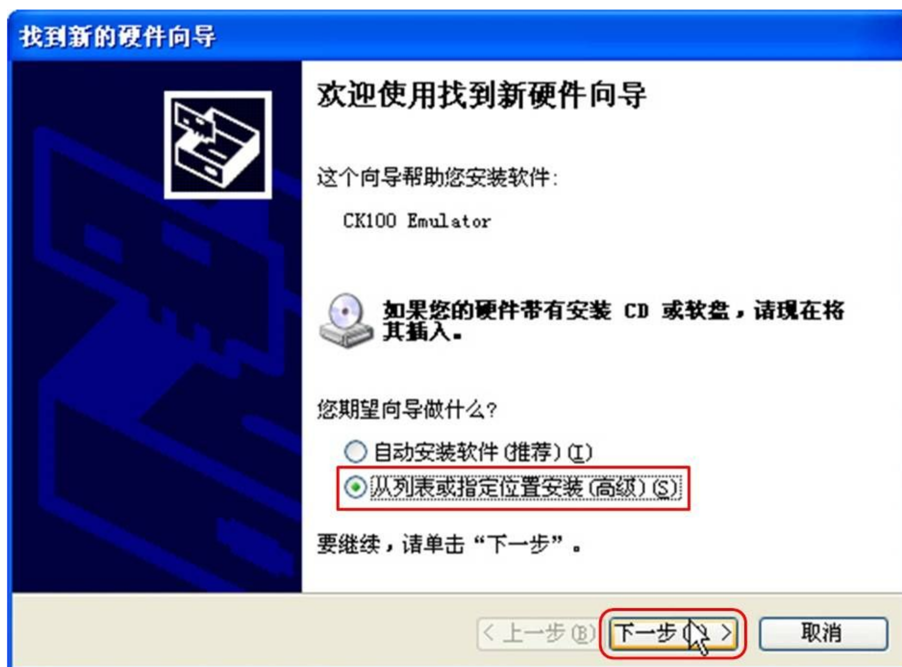


图 3.4　从列表或指定位置安装

点击图 3.5 中的【浏览】选项，进入如图 3.5 所示的界面，TKScope CK100 驱动位于 TKStudio 的安装文件夹中(本文示例为 D:\Program Files\Zhiyuan Electronic\TKStudio\DRV\TKScope\Driver\CK100 Driver\WinXP)，然后点击【确定】，再点【下一步】安装驱动。

注：TinyM0 T12/ TinyM0 T12M 开发板用到的所有驱动文件，均在 TKStudio 安装路径下的 TKScope\Driver 文件夹中。

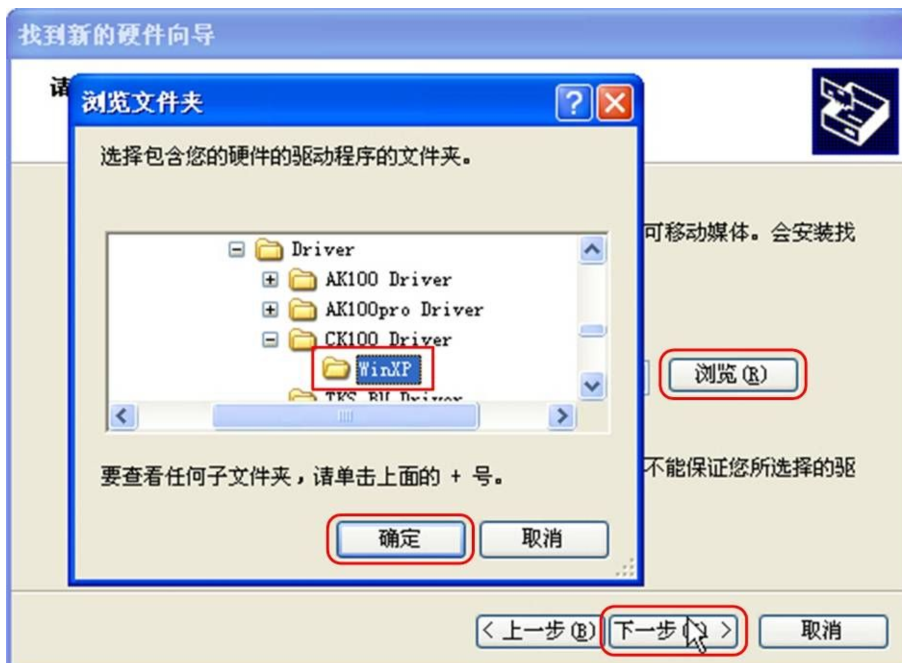


图 3.5　选择存放“CK100_Driver”文件路径

驱动安装完毕后，系统将弹出如图 3.6X 所示的对话框，单击【完成】按钮完成驱动的安装，此时，TinyM0 T12M 开发板的仿真器绿色指示灯【EMU RUN】将点亮。

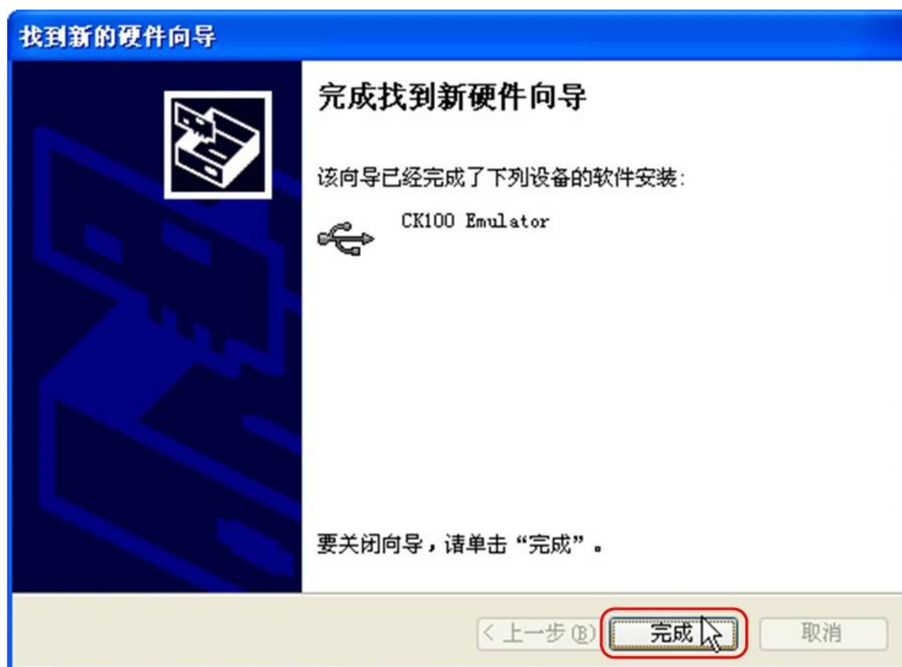


图 3.6　安装完成

系统正确安装驱动后，可以查看设备管理器里的硬件设备。右击【我的电脑】→【管理】，则弹出计算机管理界面，如图 3.7 所示。在界面左侧点击【设备管理器】，然后点击右侧窗口中的【通用串行总线控制器】前的'+'号，则可看到系统识别到的新安装的硬件设备。

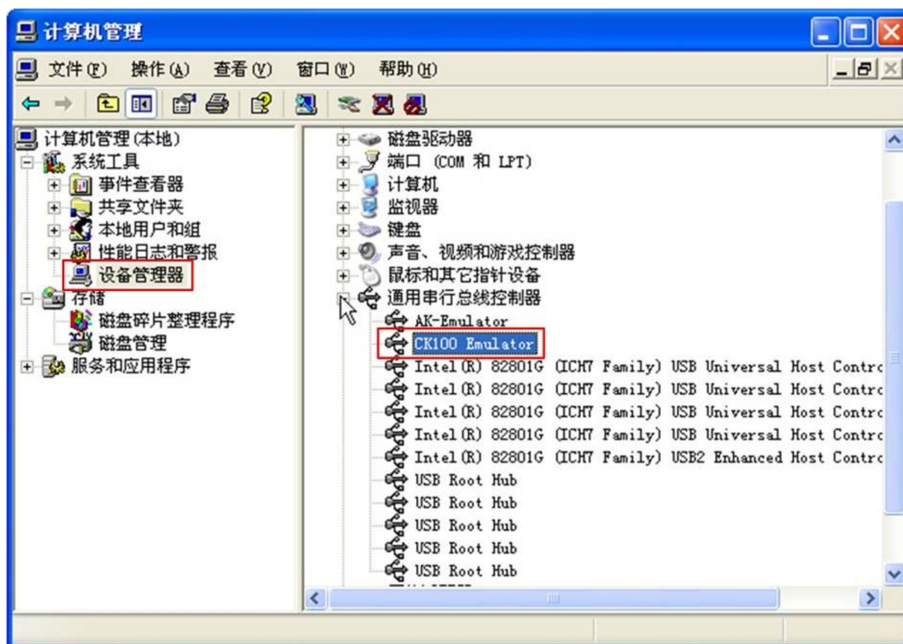


图 3.7　正确的安装新硬件结果

如果系统没有安装新硬件的驱动或驱动安装不正确，USB 设备就无法正常使用，需要重新安装驱动程序。需要点击鼠标右键，选择【更新驱动程序】选项，按照上述的过程重新安装驱动程序直到正确为止。

3.4　仿真器配置

本文以 LPC1220_GPIO_LED 工程为例(本例程请到 LPC1200 专题网页下载)，讲述如何配置 TKScope CK100 仿真器。

3.4.1 硬件连接

TinyM0 T12M 核心板上 LED1 已连接到 P1.2(P1.2 输出低电平点亮 LED1，输出高电平则熄灭 LED1)。

用 mini USB 电缆将开发板连接到 PC 机，查看开发板上仿真器绿色指示灯【EMU RUN】状态，如果【EMU RUN】闪烁一次后持续点亮，则表明仿真器硬件工作无误。否则，停止测试，检查硬件连接。

3.4.2 编译环境与仿真器设置

双击打开 LPC1220_GPIO_LED 工程文件，如图 3.8 所示。



图 3.8　打开工程文件

注：TKStudio 工程文件类型为【.XMP】，若安装 TKStudio 后，XMP 文件显示图标不为 TKS，则请重启电脑。

在 TKStudio 界面点击【工程配置】图标，如图 3.9 所示，对目标工程的参数进行配置。

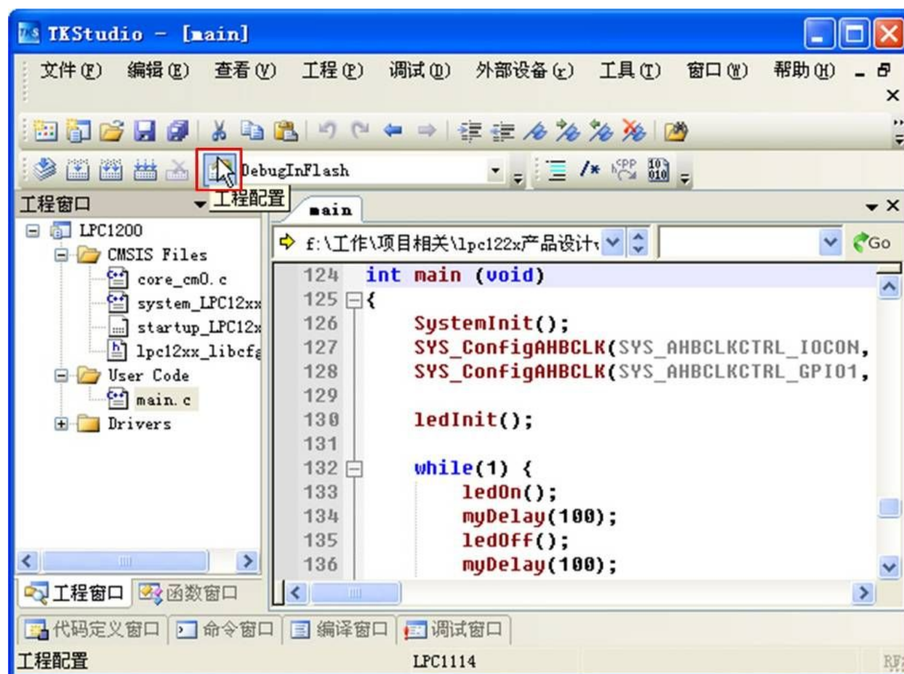


图 3.9　工程配置按钮

一般情况下，在正确选择微控制器型号(例如 LPC1227)后，编译环境设置大部分采用默认参数，如图 3.10—图 3.12 窗口所示，用户进行检查即可。

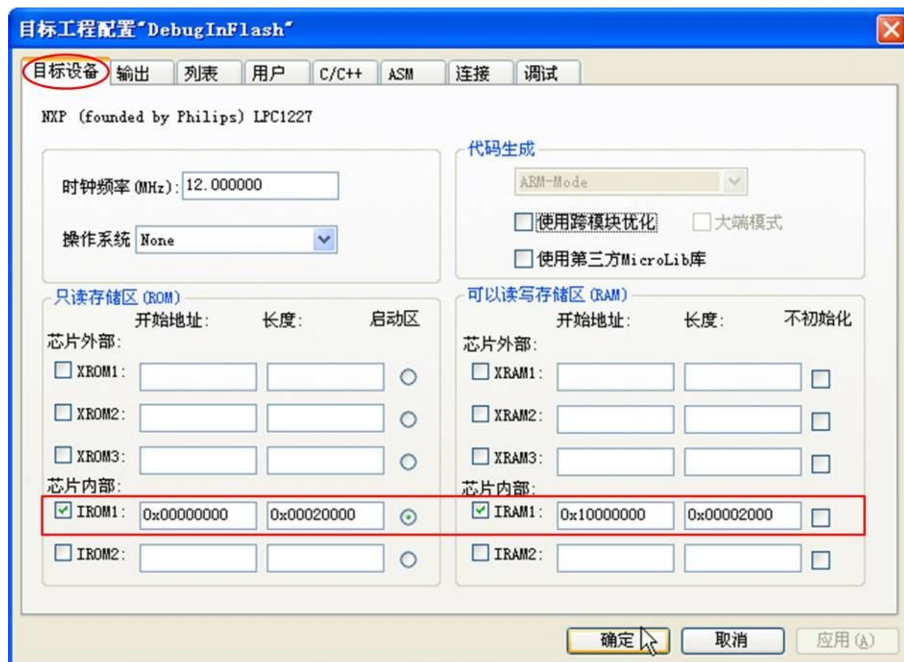


图 3.10　目标设备配置

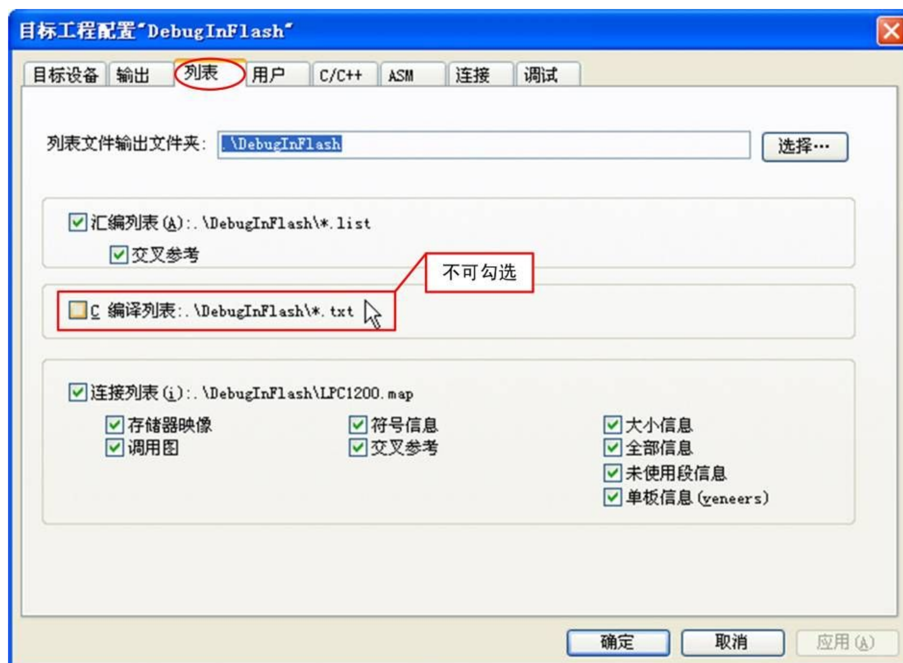


图 3.11　列表选项设置

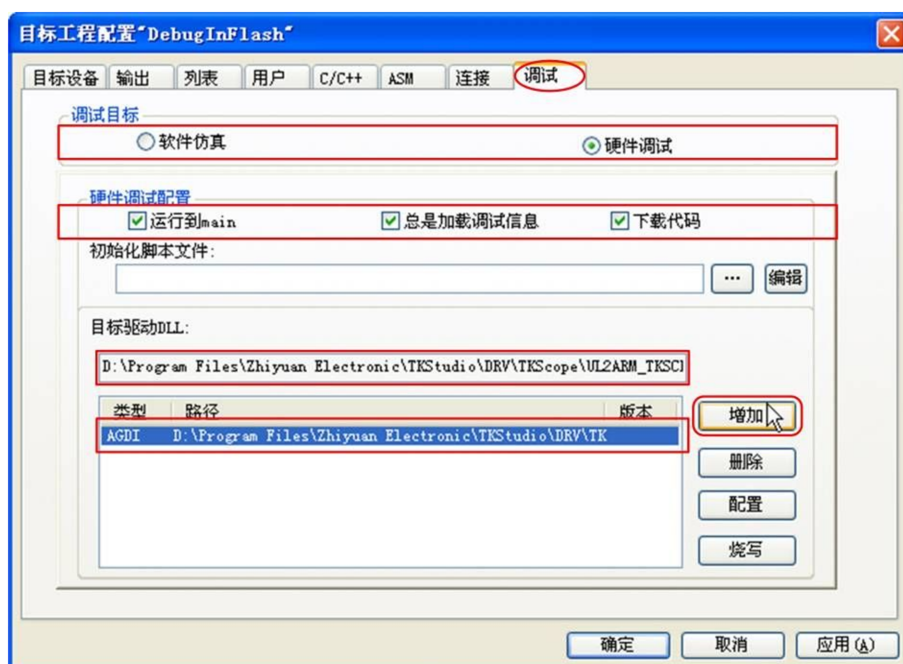


图 3.12　连接选项设置

在调试窗口，用户需要进行详细设置。

(1) 若选择硬件调试，则需要为目标添加驱动，驱动文件路径与 TKStudio 软件安装路径相关。单击【增加】按钮，弹出打开窗口如图 3.14 所示，选择驱动文件(本文示例为 D:\Program Files\Zhiyuan Electronic\TKStudio\DRV\TKScope\UL2ARM_TKSCP_DRV_ARM_for_AGDI.dll)，然后双击或单击【打开】按钮，将弹出如图 3.14 所示选择驱动路径对话框。

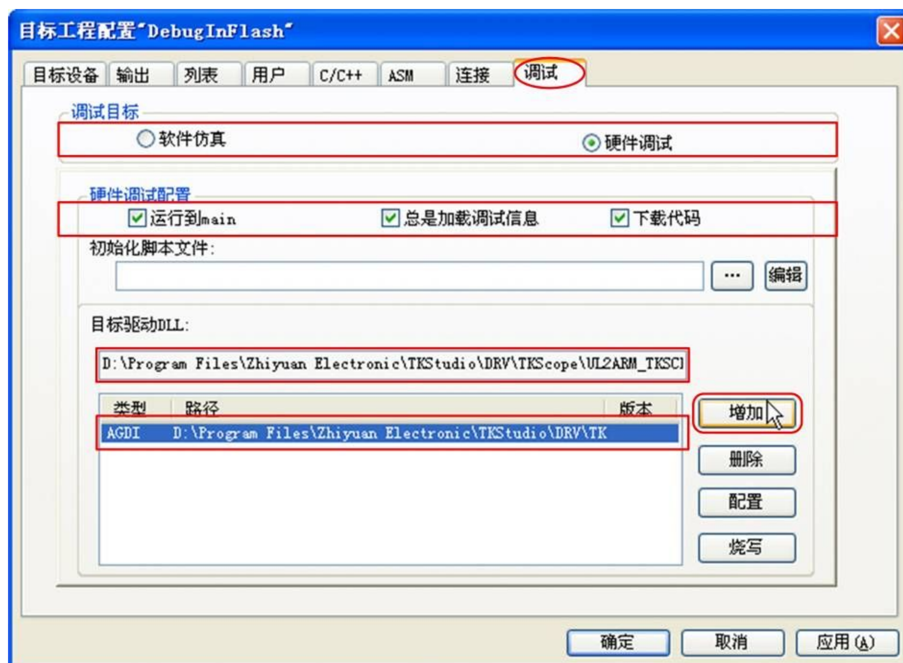


图 3.13　调试选项设置

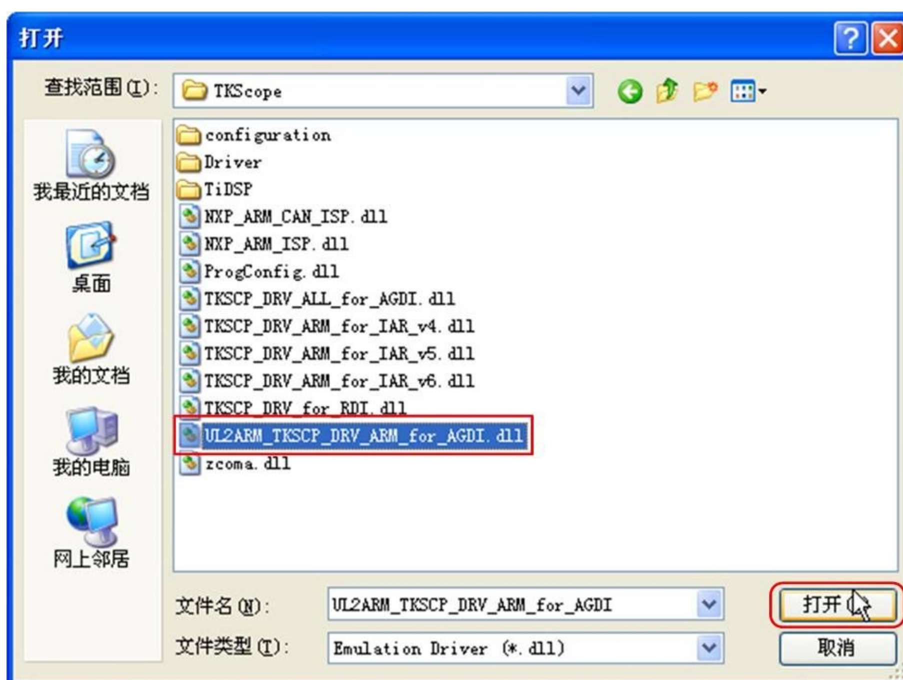


图 3.14　选择驱动

注：TinyM0 T12/ TinyM0 T12M 开发板用到的所有驱动文件，均在 TKStudio 安装路径下的 TKScope\Driver 文件夹中。

(2) 添加完 TKScope CK100 仿真器驱动后，选中该驱动程序，双击驱动名称或单击【配置】按钮，如图 3.15 所示。弹出仿真器参数设置界面，如图 3.16 所示。

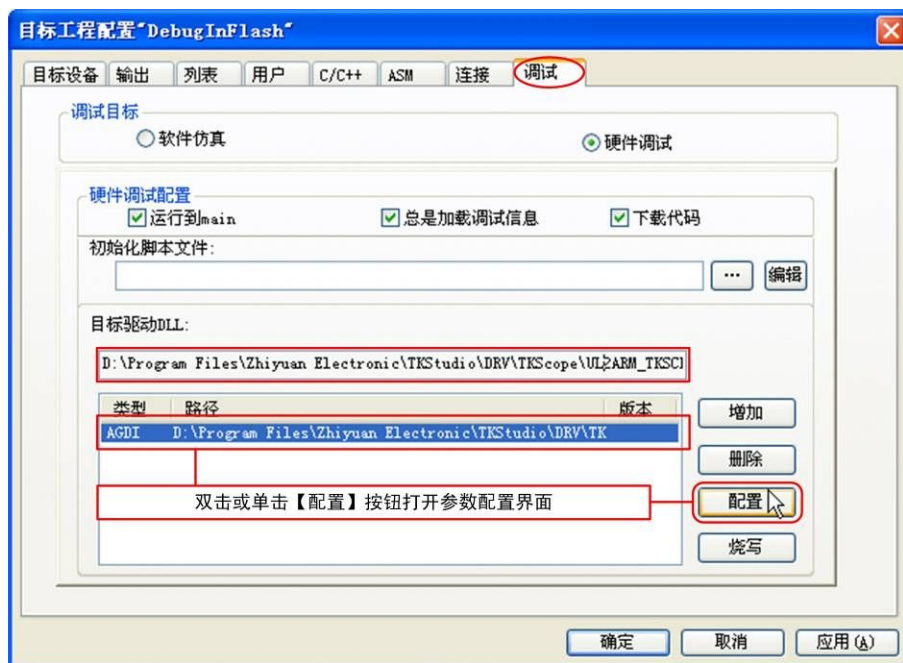


图 3.15　目标驱动配置选项

TKScope CK100 仿真器参数详细配置如下：

- 硬件选择

TKScope CK100 仿真器配置界面如图 3.16 所示，单击【硬件选择】按钮，将弹出图 3.17 所示对话框，选择仿真器型号【NXP】→【LPC1227x301】→【CK100】，完成后点击【确定】。



图 3.16　TKScope CK100 配置界面

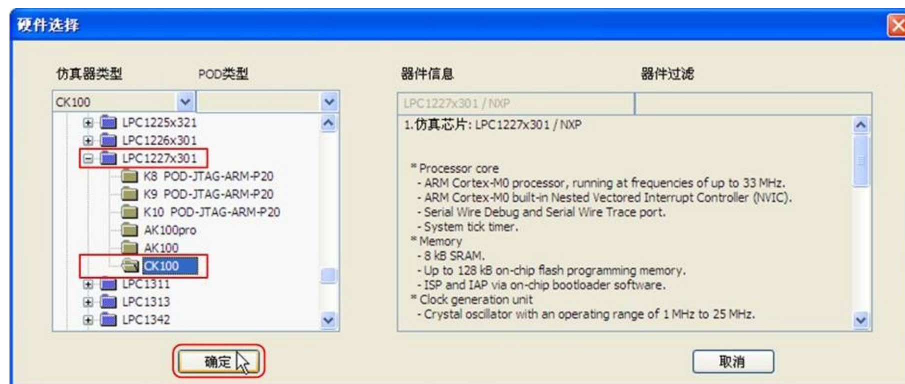


图 3.17 选择仿真器型号

然后点击图 3.16 所示界面中【搜索】按钮，若弹出如图 3.18 所示对话框，则表明仿真器硬件工作正常，单击【是(Y)】保存当前配置。



图 3.18 搜索硬件

- 主要设置

点击图 3.19 所示界面中【主要设置】按钮，将弹出如图 3.20 所示界面，需要用户逐一进行参数配置。此处必须按照本文要求进行设置，否则仿真器将无法正常工作。

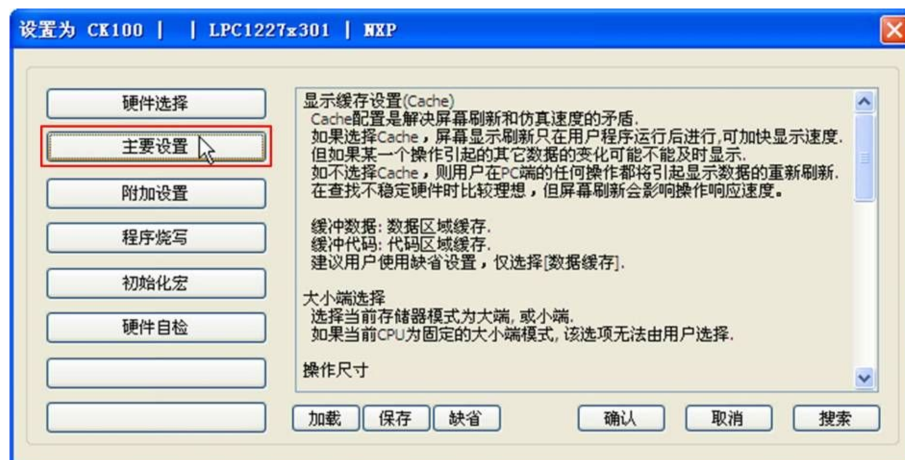


图 3.19 工程配置_主要设置



图 3.20 主要设置参数

● 附加设置

点击图 3.21 所示界面中【附加设置】按钮，将弹出如图 3.22 所示界面。由于 LPC1200 系列微控制器只支持 SWD 调试接口，所以【仿真模式】项选择 SWD 调试。



图 3.21 工程配置_附加设置



图 3.22 附加设置参数

- 程序烧写

点击图 3.23 所示界面中【程序烧写】按钮，将弹出如图 3.24 所示对话框。

TinyM0 T12M 开发板的核心控制器为 LPC1227FBD64，内部 Flash 为 128KB，若前面操作 MCU 选择正确，编程算法一栏将默认出现【LPC122x 128KB Flash】算法。



图 3.23　工程配置_程序烧写



图 3.24　程序烧写参数

注：如果接下来进行程序烧写时，Flash 编程出错则需检查此处。

- 初始化宏

点击图 3.25 所示界面中【初始化宏】按钮，将弹出如图 3.26 所示界面。此窗口采用默认设置，不需用户作修改。



图 3.25　工程配置_初始化宏

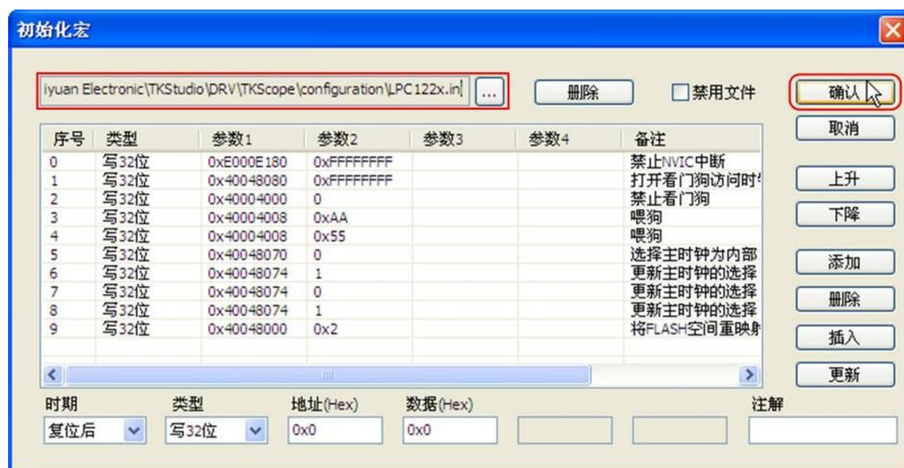


图 3.26　初始化宏参数

- 硬件自检

点击图 3.27 所示界面【硬件自检】按钮，将弹出硬件自检界面，单击【开始】按钮，若硬件初始化成功，则界面如图 3.28 所示。



图 3.27 工程配置_硬件自检



图 3.28 硬件自检界面

仿真器在硬件自检时，将读取 100000 次芯片的 ID CODE，以保证通信可靠性。由于此操作耗时较长，一般情况下，不需等待其读取完毕，可直接点击图 3.28 界面中【跳过】按钮，开始内部 SRMA 的自检操作，如图 3.29 所示。

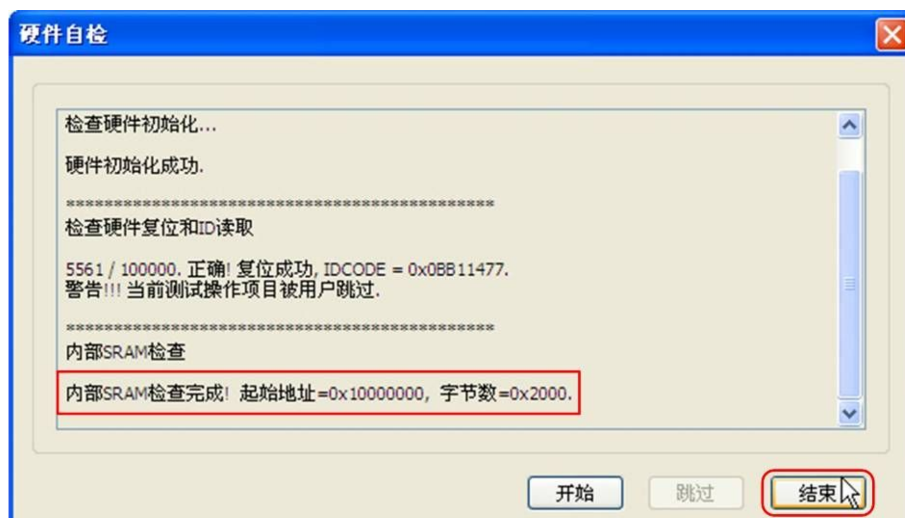


图 3.29　硬件自检完成

在芯片硬件自检完成后，点击【结束】按钮，至此即完成了 TKScope CK100 仿真器的配置工作。

注意：第一次配置完成后，点击图 3.16 界面中【保存】按钮，保存当前配置。

3.4.3 调试第一个程序

在仿真器配置完成后，返回到 TKStudio 开发环境界面，LPC1220_GPIO_LED 工程主函数的源代码如程序清单 3.1 所示。

程序清单 3.1　GPIO 控制 LED 灯闪烁

```
#include "lpc12xx_libcfg.h"

/*********************************************************************************************
  宏定义
*********************************************************************************************/
#define LED_PORT      LPC_GPIO1                                /* LED 引脚端口 P1              */
#define LED_PINS      2                                        /* LED 引脚 P1.2                */

/*********************************************************************************************
  全局变量定义
*********************************************************************************************/
static uint16_t GusGPIO1Port[7] = {                            /* GPIO1 端口引脚类型           */
    IOCON_PIO_1_0,
    IOCON_PIO_1_1,
    IOCON_PIO_1_2,
    IOCON_PIO_1_3,
    IOCON_PIO_1_4,
    IOCON_PIO_1_5,
    IOCON_PIO_1_6
};
```

```
/*********************************************************************************************************
** Function name:        myDelay
** Descriptions:         软件延时(ms)
** input parameters:     无
** output parameters:    无
** Returned value:       无
*********************************************************************************************************/
void myDelay (uint32_t ulTime)
{
    uint32_t i = 0;

    while (ulTime--) {
        for (i = 0; i < 5000; i++);
    }
}

/*********************************************************************************************************
** Function name:        ledOn
** Descriptions:         点亮 LED(输出低电平)
** input parameters:     无
** output parameters:    无
** Returned value:       无
*********************************************************************************************************/
void ledOn (void)
{
    GPIO_SetLowLevel(LED_PORT, LED_PINS, ENABLE);          /* 输出低电平，点亮 LED           */
}

/*********************************************************************************************************
** Function name:        ledOff
** Descriptions:         熄灭 LED(输出高电平)
** input parameters:     无
** output parameters:    无
** Returned value:       无
*********************************************************************************************************/
void ledOff (void)
{
    GPIO_SetHighLevel(LED_PORT, LED_PINS, ENABLE);         /* 输出高电平，熄灭 LED           */
}

/*********************************************************************************************************
** Function name:        ledInit
** Descriptions:         LED 初始化
```

```
** input parameters:        无
** output parameters:       无
** Returned value:          无
*********************************************************************************************/
void ledInit (void)
{
    IOCON_PIO_CFG_Type PIO_mode;

    IOCON_StructInit(&PIO_mode);                              /*  初始化端口模式                  */
    PIO_mode.type = GusGPIO1Port[LED_PINS];
    IOCON_SetFunc(&PIO_mode);                                 /*  设置 LED 引脚为 GPIO 功能         */

    GPIO_SetDir(LED_PORT, LED_PINS, GPIO_DIR_OUTPUT); /*  设置 LED 引脚为输出                 */

    ledOff();                                                 /*  熄灭 LED                        */
}

/*********************************************************************************************
** Function name:      main
** Descriptions:       主函数(函数入口)
**     串口参数：      UART1 波特率 9600、8 个数据位、1 个停止位、无奇偶校验位
**     跳线连接：      P0.8(RXD1)、P0.9(TXD1)通过 232 电平转换芯片分别连接到 PC 机串口的 TXD、RXD；
**     操作方法：      打开串口调试软件，运行程序，观察显示窗口及 LED1 状态
**     现      象：    LED1 以约 200ms 为周期闪烁
** input parameters:   无
** output parameters:  无
** Returned value:     无
*********************************************************************************************/
int main (void)
{
    SystemInit();                                                 /*  系统时钟初始化              */
    SYS_ConfigAHBCLK(SYS_AHBCLKCTRL_IOCON, ENABLE); /* IOCON 模块使能                   */
    SYS_ConfigAHBCLK(SYS_AHBCLKCTRL_GPIO1, ENABLE); /* GPIO1 时钟使能                   */

    ledInit();                                                    /* LED 初始化                   */

     while(1) {
         ledOn();
         myDelay(100);
         ledOff();
         myDelay(100);
    }
}
```

(1) 选择 DegugInFlash，程序代码将下载到片内 Flash 中，然后点击编译链接工具条对

LPC1220_GPIO_LED 工程进行编译，如图 3.30 所示。

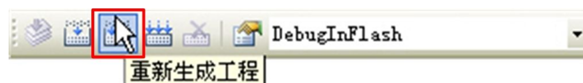


图 3.30　重新编译文件

(2) 编译完成后，点击 Debug 调试工具，如图 3.31 所示，启动 TKStudio 调试。在进入调试环境前，需要先对 Flash 进行编程，如图 3.32 所示。



图 3.31　进入调试环境



图 3.32　Flash 编程

(3) 进入调试环境后，光标将停止在 main 函数，如图 3.33 所示。

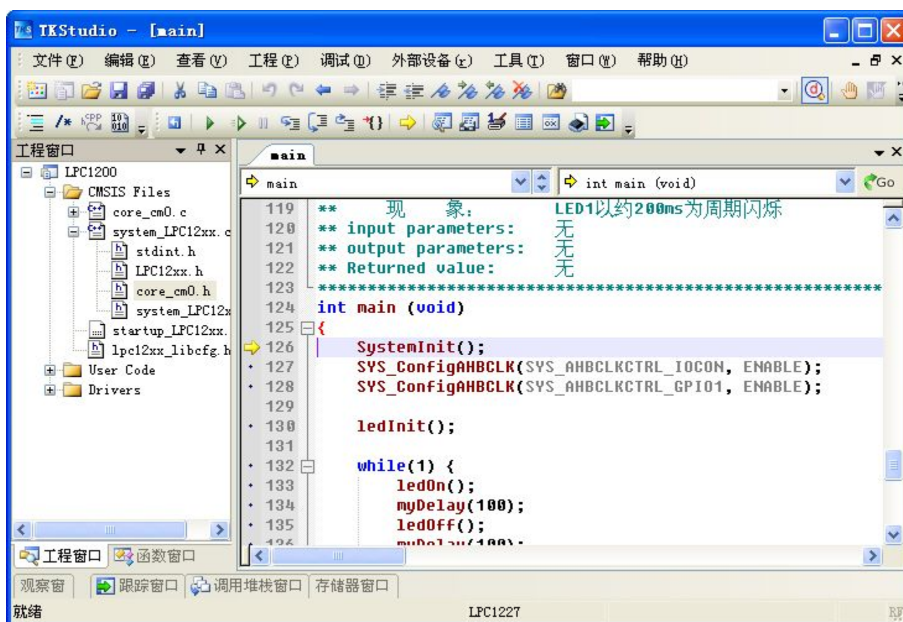


图 3.33　程序进入调试环境后现象

此时用户可点击如图 3.34 所示【显示/隐藏反汇编窗口】按钮打开反汇编确认代码是否已经下载到芯片中。如果代码下载成功，则可得到如图 3.34 所示的结果，可以看到 C 代码中间插入了汇编代码，用户可以进行分析。但如果反汇编的结果全部是“0”或者“F”，或者全部汇编语句都是相同的，则说明程序没有下载进去，需要重新下载。

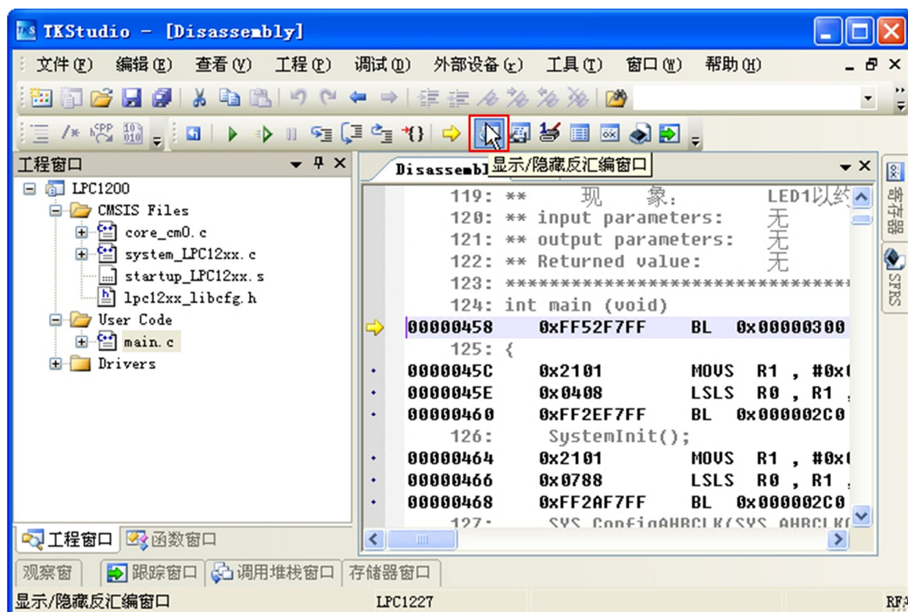


图 3.34 查看反汇编

(4) 点击全速运行图标，如图 3.35 所示，即可看到开发板上 LED1 灯周期性闪烁。



图 3.35 全速运行图标

(5) 若停止程序运行，则需点击如图 3.36 所示的【停止】按钮。



图 3.36 停止运行

至此，已完整地讲述了从安装软件到调试程序的步骤、方法。

4. K-Flash

4.1 概述

K-Flash 是由广州致远电子有限公司开发，集成在 TKStudio 开发环境内的在线烧写工具，主要用于 Flash 烧写，具有文件烧写与校验、数据擦除和数据读取等功能。K-Flash 工具路径如图 4.1 所示，用户也可直接从 TKStudio 安装目录下打开 K-Flash。



图 4.1 TKStudio 中 K-Flash 路径

K-Flash 具有以下特点：

- 工程化配置，操作简单、方便、快捷；
- 支持 TKScope 全系列通用仿真器；
- 支持 ARM、51、AVR 等多种内核；
- 支持一次烧写多个文件，操作更快捷；
- 支持 bin 文件、hex 文件、elf 文件等多种类型的文件；
- 支持片内 Flash 和片外 Flash 烧写(NOR/NAND/SPI 等)；
- 具备烧写校验功能，验证烧写是否正确，可详细查看每一处差异；
- 多种 Flash 烧写操作，可以进行擦除和读取等操作；
- 具有独立的数据擦除功能，可指定扇区进行擦除，操作更简捷；
- 支持 ISP 擦除操作；
- 具有独立的数据读取功能，可指定地址范围进行读取，支持一步到位打开读取出的数据；
- 保存配置到工程文件中，免除繁琐重复的配置操作；
- 支持各大半导体公司的芯片烧写，内置多种芯片的默认配置；
- ARM 内核烧写算法文件公开，用户可自行添加 Flash 算法。

4.2 K-Flash 操作

在 TKStudio 集成开发环境下运行 K-Flash，主界面如图 4.2 所示。



图 4.2　K-Flsh 主界面

使用 K-Flash 进行烧写，擦除，读取等操作之前，必须对设备进行配置，不同内核的芯片设备配置也不同。本文只介绍 ARM 芯片的设备配置，AVR、51 等其它芯片的设备配置请参考 TKScope 仿真器用户手册。

1.　设备配置

点击图 4.2 中的【设备配置】按钮，将弹出一个对话框，如图 4.3 所示。



图 4.3　设备配置

点击图 4.3 中【硬件选择】按钮进入如图 4.4 所示的界面，此时必须正确选择当前烧写的芯片型号和仿真器类型，配置方法与本文 3.4 小节的仿真器配置完全相同。



图 4.4 选择仿真器型号

除需设置【硬件选择】一项之外，【程序烧写】一项也需用户对其进行设置，点击图 4.5 中的【程序烧写】按钮将弹出界面如图 4.6 所示。



图 4.5 程序烧写

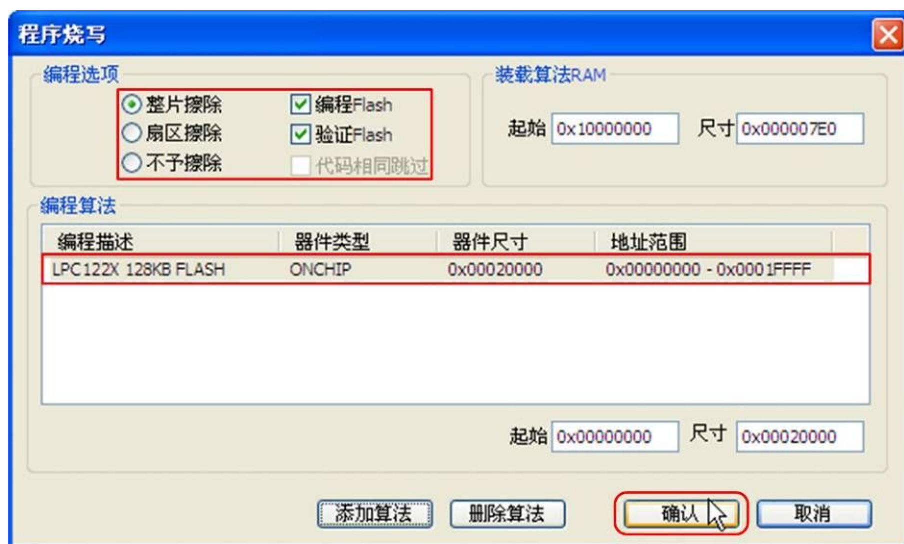


图 4.6 程序烧写参数设置

(1) 编程选项

- 整片擦除：程序下载到 Flash 前擦除全部 Flash 空间；
- 扇区擦除：根据下载需要擦除相应的扇区空间；
- 不予擦除：不进行 Flash 擦除。

(2) 装载算法 RAM

- 起始：可用的 RAM 起始地址，用户一般不需要修改；
- 尺寸：当前 RAM 的尺寸，给定的数值可以满足用户需要，适当减小可加快烧写速度。

(3) 编程算法

- 起始：当前算法文件描述器件的起始地址；
- 尺寸：当前算法文件描述器件的尺寸。

(4) 编程算法文件的操作

- 添加算法：添加用户指定的编程算法文件。具体方法：点击【添加算法】按钮，出现一个打开对话框，找到并选择要添加的算法(本文示例为 D:\Program Files\Zhiyuan Electronic\TKStudio\DRV\TKScope\configuration)，点击【打开】按钮，算法将添加到编程算法列表框中。如果要添加多个算法，重复上述步骤，一一添加即可(最多可添加五个算法)；
- 删除算法：删除当前选择的算法。该操作对系统默认的算法无效。

2. 烧写

设备在配置完成后即可进行烧写、检验、擦除和读取等操作。点击如图 4.2 所示中烧写功能区的【打开】按钮，添加烧写的目标文件(HEX 或 BIN，BIN 文件还需要设置目标地址)，然后点击【烧写】按钮后即可下载代码，同时将弹出一个程序烧写的进度条，如图 4.7 所示。



图 4.7 程序烧写

3. 校验

烧写完成后，点击【校验】按钮，开始对烧写的数据进行校验，如果直接点击【烧写校验】按钮，则烧写之后将自动进行数据校验。

5. 串口调试助手

5.1 概述

TKS_COM 是由广州致远电子有限公司开发，集成在 TKStudio 开发环境内的串口调试工具。TKS_COM 除具备一般串口调试工具的基本功能外，还具有对多串口的实时监控、桥接、多播和数据过滤等功能。TKS_COM 工具路径如图 5.1 所示。



图 5.1 TKStudio 中 TKS_COM 路径

TKS_COM 串口调试助手功能特点：

- 能够自动识别当前 PC 机上的所有串口；
- 保存配置信息，方便下次使用；
- 发送接收数据，支持中文、ASCII 和 HEX 三种格式；
- 支持字符串的单次发送和循环发送；
- 支持对接收数据实时保存；
- 支持发送文件；
- 支持终端调试；
- 能够对多串口进行实时监控，并提供任意串口对之间的桥接功能；
- 支持串口广播以及定时广播；
- 支持对接收数据进行过滤操作。

5.2 TKS_COM 操作

在 TKStudio 集成开发环境下运行 TKS_COM 串口调试助手软件，主界面如图 5.2 所示。

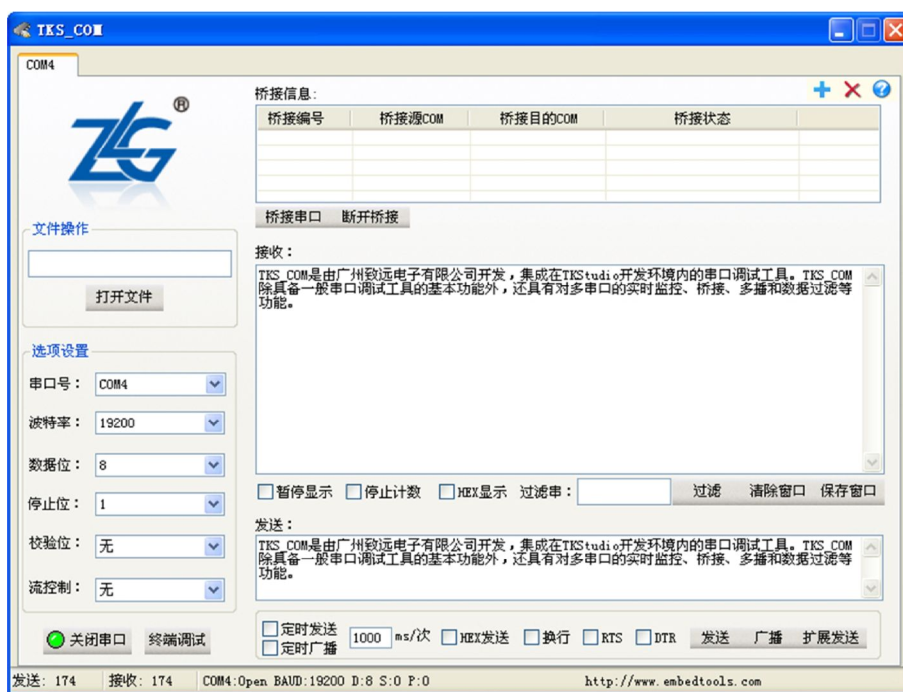


图 5.2　串口调试助手主界面

1.　打开/关闭串口

点击图 5.2 中 TKS-COM 主界面左下角的【打开串口】按钮即可打开已选定的串口，同时【打开串口】字样将变为【关闭串口】，再次点击后将关闭相应的串口。

若第一次打开 TKS-COM 软件，会在 TKS-COM 的同级目录下生成一个名为【TKS-COM.ini】的配置文件，第二次打开 TKS-COM 时将采用默认的配置信息。

2.　发送文件

TKS-COM 支持文件发送功能，一定程度上提高了批量数据发送的速度。

点击图 5.2 中 TKS-COM 主界面中的【打开文件】按钮，选择要发送的文件，然后分别设置数据包的大小与时间间隔，点击【发送】即可开始发送目标文件，这时将显示文件发送的进度等参数，示例应用如图 5.3 所示。

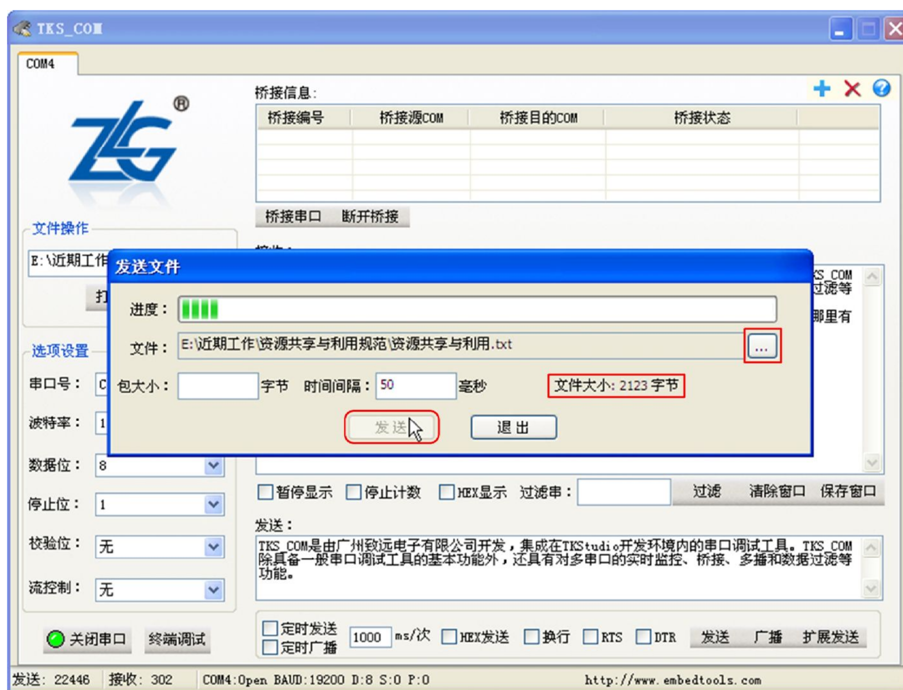


图 5.3　发送文件

3.　发送与接收数据

在 TKS-COM 发送窗口编辑完数据后，点击【发送】按钮即可开始数据传输，接收区用于显示串口接收到的数据，若将【过滤】按钮按下，则接收窗口将不显示过滤字符串如图 5.4 所示为同一台 PC 机上串口自发自收数据显示示例。

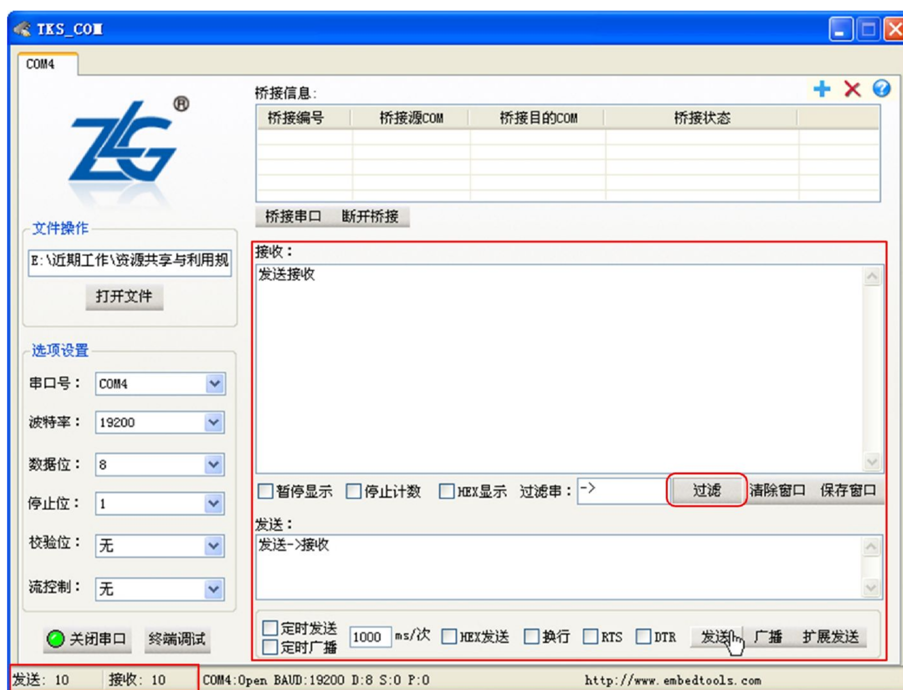


图 5.4　发送与接收数据

对于发送和接收的方式可分别设置，具体如下：

- 可以定时发送或定时方播；

- 可以设置发送或接收的数据格式，以 ASCII 或 Hex 格式发送或接收；
- 可以进行多行发送；
- RTS/DTR 选项可以实现发送数据的电平控制；
- “广播”可以将发送区的数据广播到所有打开的串口；
- “扩展发送”可以进行多字符串发送。

6. 免责声明

广州致远电子有限公司随附提供的软件或文档资料旨在提供给您(本公司的客户)使用，仅限于且只能在本公司制造或销售的产品上使用。

该软件或文档资料为本公司和/或其供应商所有，并受适用的版权法保护，版权所有。如有违反，将面临相关适用法律的刑事制裁，并承担违背此许可的条款和条件的民事责任。

本公司保留在不通知读者的情况下，修改文档或软件相关内容的权利，对于使用中所出现的任何效果，本公司不承担任何责任。

该软件或文档资料“按现状”提供，不提供保证，无论是明示的、暗示的还是法定的保证。这些保证包括(但不限于)对出于某一特定目的应用此软件的适销性和适用性默示的保证。在任何情况下，公司不会对任何原因造成的特别的、偶然的或间接的损害负责。